

IJNS

**International Journal
of Network Security**



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

Vol. 19, No. 5 (Sept. 2017)

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)

Çetin 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.

1. A Survey of Blockchain Security Issues and Challenges
Iuon-Chang Lin and Tzu-Chun Liao 653-659
2. Efficient Feature Selection Technique for Network Intrusion Detection System Using Discrete Differential Evolution and Decision
Ebenezer Popoola, Aderemi Oluyinka Adewumi 660-669
3. GPS Spoofing Detection Based on Decision Fusion with a K-out-of-N Rule
Minhong Sun, Yuan Qin, Jianrong Bao, Xutao Yu 670-674
4. Design and Implementation of an Intrusion Prevention System Inspired Immune Systems
Yousef Farhaoui 675-683
5. Securing Portable Document Format File Using Extended Visual Cryptography to Protect Cloud Data Storage
K. Brindha and N. Jeyanthi 684-693
6. Design and Implementation of Secure Remote e-Voting System Using Homomorphic Encryption
Ihsan Jabbar and Saad Najim Alsaad 694-703
7. A Key-Policy Attribute-based Encryption Scheme for General Circuit from Bilinear Maps
Peng Hu, Haiying Gao 704-710
8. An Anti-Phishing Password Authentication Protocol
Pramote Kuacharoen 711-719
9. Role-based Access Control for Body Area Networks Using Attribute-based Encryption in Cloud Storage
Ye Tian, Yanbin Peng, Gaimei Gao, Xinguang Peng 720-726
10. A Comparative Study on Feature Selection Method for N-gram Mobile Malware Detection
Mohd Zaki Mas'ud, Shahrin Sahib, Mohd Faizal Abdollah, Siti Rahayu Selamat, Choo Yun Huoy 727-733
11. Discriminating Flash Events from DDoS Attacks: A Comprehensive Review
Sunny Behal, Krishan Kumar, Monika Sachdeva 734-741
12. Information Security Risk Management Framework for University Computing Environment
Umesh Kumar Singh, Chanchala Joshi 742-751
13. Speech Perceptual Hashing Authentication Algorithm Based on Spectral Subtraction and Energy to Entropy Ratio
Qiu-Yu Zhang, Wen-Jin Hu, Si-Bin Qiao, and Yi-Bo Huang 752-760
14. An Unsupervised Method for Detection of XSS Attack
Swaswati Goswami, Nazrul Hoque, Dhruva K. Bhattacharyya, Jugal Kalita 761-775
15. Medical Image Encryption Scheme Based on Arnold Transformation and ID-AK Protocol
Osman Wahballa, Abubaker Wahaballa, Fagen Li, Idris Ibn Idris and Chunxiang Xu 776-784

16. An Enhanced Anonymous Password-based Authenticated Key Agreement Scheme with Formal Proof
Min Wu, Jianhua Chen, Ruibing Wang 785-793
17. A General Formal Framework of Analyzing Selective Disclosure Attribute-Based Credential Systems
Caimei Wang, Yan Xiong, Wenjuan Cheng, Wenchao Huang, Huihua Xia, Jianmeng Huang 794-803
18. A New Security Cloud Storage Data Encryption Scheme Based on Identity Proxy Re-encryption
Caihui Lan, Haifeng Li, Shoulin Yin, and Lin Teng 804-810
19. Policy-based Signatures for Predicates
Fei Tang, Yousheng Zhou 811-822
20. A New Way to Prevent UKS Attacks Using Hardware Security Chips
Qianying Zhang, Zhiping Shi 823-831
21. Whirlwind: A New Method to Attack Routing Protocol in Mobile Ad Hoc Network
Luong Thai Ngoc and Vo Thanh Tu 832-838
22. Distinguishing Medical Web Pages from Pornographic Ones: An Efficient Pornography Websites Filtering Method
Jyh-Jian Sheu 839-850
23. Privacy-preserving Similarity Sorting in Multi-party Model
Yifei Yao, Fanhua Yu 851-857
24. An Improved Dual Image-based Reversible Hiding Technique Using LSB Matching
Yu-Lun Wang, Jau-Ji Shen, Min-Shiang Hwang 858-862

A Survey of Blockchain Security Issues and Challenges

Iuon-Chang Lin^{1,2} and Tzu-Chun Liao²

(Corresponding author: Iuon-Chang Lin)

Department of Photonics and Communication Engineering, Asia University¹
500, Lioufeng Rd., Wufeng, Taichung 41354, Taiwan

Department of Management Information Systems, National Chung Hsing University²
145 Xingda Rd., South Dist., Taichung City 402, Taiwan

(Email: corresponding_iclin@nchu.edu.tw)

(Invited Jan. 12, 2017)

Abstract

Blockchain technologies is one of the most popular issue in recent years, it has already changed people's lifestyle in some area due to its great influence on many business or industry, and what it can do will still continue cause impact in many places. Although the feature of blockchain technologies may bring us more reliable and convenient services, the security issues and challenges behind this innovative technique is also an important topic that we need to concern.

Keywords: Blockchain; Smart Contracts; Security

1 Introduction

Bitcoin is the first application of blockchain, it's a kind of digital currency based on blockchain technologies, using for trade things on the internet like money as we do in the real world. Because the success of Bitcoin, people now can utilize blockchain technologies in many field and service, such as financial market, IOT, supply chain, voting, medical treatment and storage.

But as we use these tools or services in our daily life, cybercriminals also get opportunity to engage in cybercrime [16, 18]. For example, 51% attacks is a classic security issue in Bitcoin that hacker try to take control the system's mechanism, using the same technology base.

In this paper, we will have a quick study about what is blockchain in Section 2, then we'll discuss different application in blockchain and what service do they offer in Section 3, at the end, we shall talk about the security issues and those challenges we need to overcome in Section 4. The paper is concluded in Section 5.

2 The Concept of Blockchain

Blockchain technologies is not just only single one technique, but contains Cryptography, mathematics, Algorithm and economic model, combining peer-to-peer networks and using distributed consensus algorithm to solve traditional distributed database synchronize problem, it's an integrated multi- field infrastructure construction [5, 6, 15].

The blockchain technologies composed of six key elements.

Decentralized. The basic feature of blockchain, means that blockchain doesn't have to rely on centralized node anymore, the data can be record, store and update distributedly.

Transparent. The data's record by blockchain system is transparent to each node, it also transparent on update the data, that is why blockchain can be trusted.

Open Source. Most blockchain system is open to everyone, record can be check publicly and people can also use blockchain technologies to create any application they want.

Autonomy. Because of the base of consensus, every node on the blockchain system can transfer or update data safely, the idea is to trust form single person to the whole system, and no one can intervene it.

Immutable. Any records will be reserved forever, and can't be changed unless someone can take control more than 51% node in the same time.

Anonymity. Blockchain technologies solved the trust problem between node to node, so data transfer or even transaction can be anonymous, only need to know the person's blockchain address.

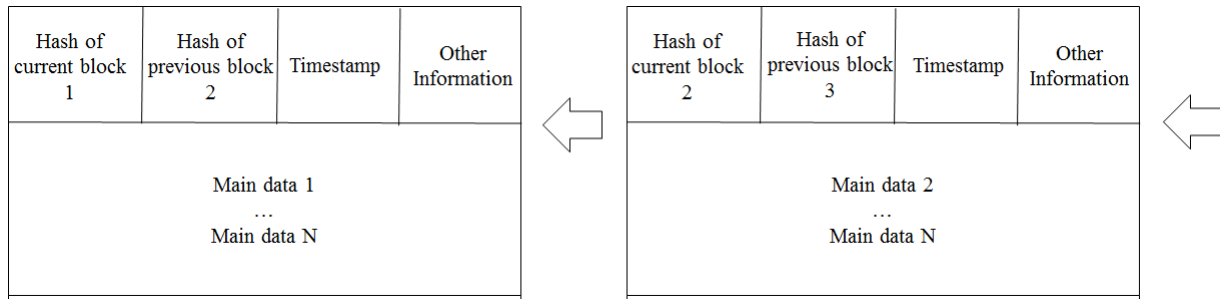


Figure 1: The structure of block chain

2.1 How Blockchain Works?

The main working processes of blockchain are as follows:

- 1) The sending node records new data and broad casting to network.
- 2) The receiving node checked the message from those data which it received, if the message was correct then it will be stored to a block.
- 3) All receiving node in the network execute proof of work (PoW) or proof of stake (PoS) algorithm to the block.
- 4) The block will be stored into the chain after executing consensus algorithm, every node in the network admit this block and will continuously extend the chain base on this block.

2.2 The Structure of Blockchain

Generally in the block, it contains main data, hash of previous block, hash of current block, timestamp and other information. Figure 1 shows the structure of block.

Main data. Depending on what service is this blockchain applicatte, for example: transaction records, bank clearing records, contract records or IOT data record.

Hash. When a transaction executed, it had been hash to a code and then broadcast to each node. Because it could be contained thousands of transaction records in each node’s block, blockchain used Merkle tree function to generate a final hash value, which is also Merkle tree root. This final hash value will be record in block header (hash of current block), by using Merkle tree function, data transmission and computing resources can be drastically reduced.

Timestamp. Time of block generated.

Other Information. Like signature of the block, Nonce value, or other data that user define.

2.3 How to Get Consensus?

Consensus function is a mechanism that make all blockchain nodes have agreement in same message, can make sure the latest block have been added to the chain correctly, guarantee the message that stored by node was the same one and won’t happened “fork attack”, even can protect from malicious attacks.

2.4 Proof of Work (PoW)

A proof of work is a piece of data which is difficult (costly or time-consuming) to produce but easy for others to verify and which satisfies certain requirements. Producing a proof of work can be a random process with low probability so that a lot of trial and error is required on average before a valid proof of work is generated. Bitcoin uses the Hashcash proof of work system.

When calculating PoW, it’s called “mining”. Each block has a random value called “Nonce” in block header, by changing this nonce value, PoW have to generate a value that makes this block header hash value less than a “Difficulty Target” which has already been set up. Difficulty means how much time it will take when the node calculating hash value less than target value.

In order for a block to be accepted by network participants, miners must complete a proof of work which covers all of the data in the block. The difficulty of this work is adjusted so as to limit the rate at which new blocks can be generated by the network to one every 10 minutes. Due to the very low probability of successful generation, this makes it unpredictable which worker computer in the network will be able to generate the next block [1, 7].

2.5 Proof of Stake (PoS)

Because Proof of Work method will cause a lot of electricity power and computing power be wasted, Proof of Stake doesn’t need expensive computing power. With Proof of Stake, the resource that’s compared is the amount of Bitcoin a miner holds - someone holding 1% of the Bitcoin can mine 1% of the “Proof of Stake blocks” [12].

A Proof of Stake method might provide increased protection from a malicious attack on the network. Additional protection comes from two sources:

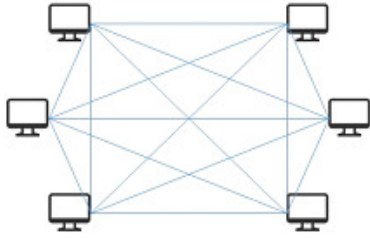


Figure 2: Public blockchain

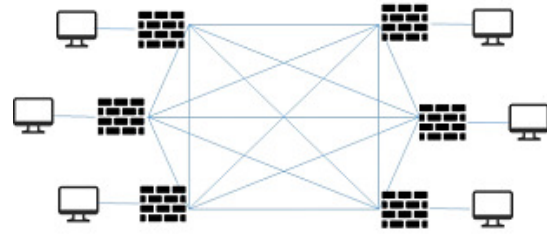


Figure 4: Private blockchain

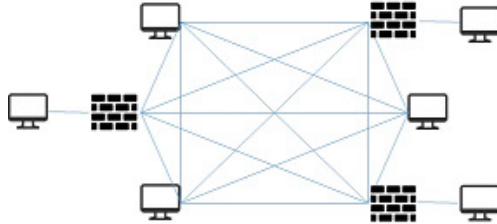


Figure 3: Consortium blockchain

- 1) Executing an attack would be much more expensive.
- 2) Reduced incentives for attack. The attacker would need to own a near majority of all bitcoin. Therefore, the attacker suffer severely from his own attack.

2.6 Type of Blockchain

Blockchain technologies can be roughly divided into three types.

- 1) **Public blockchain:** Everyone can check the transaction and verify it, and can also participate the process of getting consensus. Like Bitcoin and Ethereum are both Public Blockchain. Figure 2 shows public blockchain.
- 2) **Consortium blockchains:** It means the node that had authority can be choose in advance, usually has partnerships like business tobusiness, the data in blockchain can be open or private, can be seen as Partly Decentralized. Like Hyperledger and R3CEV are both consortium blockchains. Figure 3 shows consortium blockchains.
- 3) **Private blockchain:** Node will be restricted, not every node can participate this blockchain, has strict authority management on data access. Figure 4 shows private blockchain.

No matter what types of blockchain is, it both has advantage. Sometimes we need public blockchain because its convenience, but sometimes we maybe need private control like consortium blockchains or private blockchain, depending on what service we offer or what place we use it.

3 Application of Blockchain Technologies

Blockchain technologies can be using in many area, not only in financial application, but also in others industries.

3.1 Digital Currency: Bitcoin

Bitcoin's data structure and transaction system was built by blockchain technologies, makes Bitcoin became a digital currency and online payment system. By using encrypted technique, funds transfer can be achieved and doesn't need to rely on central bank. Bitcoin used public keys address sending and receiving bitcoin, recorded the transaction and the personal ID was anonymous. The process of transaction confirm needs other user's computing power to get consensus, and then records the transaction to network.

3.2 Smart Contract: Ethereum

Smart Contract is a digital contract that controls user's digital assets, formulating the participant's right and obligation, will automatically execute by computer system. It's not only just a computer procedure, it can be seen as one of a contract participants, will response to message what it receive and store the data, it can also send message or value to outside. Smart Contract is just like a person can be trusted, can hold the assets temporarily and will follow the order which has already been program [13].

Ethereum is an open source blockchain platform combining Smart Contract, offering decentralized virtual machine to handle the contract, by using its digital currency called ETH, people can create many different services, applications or contracts on this platform [21].

3.3 Hyperledger

Hyperledger is an open source blockchain platform, started in December 2015 by the Linux Foundation, to support blockchain-based distributed ledgers. It is focused on ledgers designed to support global business transactions, including major technological, financial, and supply chain companies, with the goal of improving many aspects of performance and reliability. The project aims

to bring together a number of independent efforts to develop open protocols and standards, by providing a modular framework that supports different components for different uses. This would include a variety of blockchains with their own consensus and storage models, and services for identity, access control, and contracts.

3.4 Other Applications

There still have many use case of blockchain technologies, like protection of Intellectual property, traceability in supply chain, identity certification, insurance, international payments, IOT, patient’s privacy in medical treatment or prediction market [14, 20].

4 Security Issues and Challenges

So far, blockchain has been gotten many attention in different area, however, it also exists some problems and challenges needs to face it [2, 9].

4.1 The Majority Attack (51% Attacks)

With Proof of Work, the probability of mining a block depends on the work done by the miner (e.g. CPU/GPU cycles spent checking hashes). Because of this mechanism, people will want to join together in order to mining more blocks, and become “mining pools”, a place where holding most computing power. Once it hold 51% computing power, it can take control this blockchain. Apparently, it cause security issues [3, 4].

If someone has more than 51% computing power, then he/she can find Nonce value quicker than others, means he/she has authority to decide which block is permissible. What it can do is:

- 1) Modify the transaction data, it may cause double-spending attack [11, 17].
- 2) To stop the block verifying transaction.
- 3) To stop miner mining any available block.

A majority attack was more feasible in the past when most transactions were worth significantly more than the block reward and when the network hash rate was much lower and prone to reorganization with the advent of new mining technologies [8].

4.2 Fork Problems

Another issue is fork problem. Fork problem is related to decentralized node version, agreement when the software upgrade. It is a very important issue because it involving a wide range in blockchain.

- Types of Forks

When the new version of blockchain software published, new agreement in consensus rule also changed

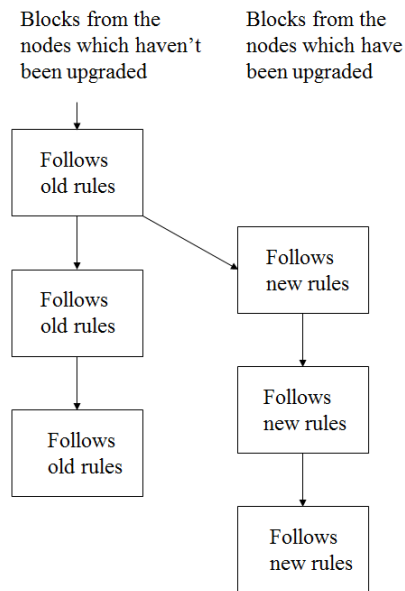


Figure 5: Hard Fork

to the nodes. Therefore, the nodes in blockchain network can be divided into two types, the New Nodes and the Old Nodes. So here come four situations:

- 1) The new nodes agree with the transaction of block which is sending by the old nodes.
- 2) The new nodes don’t agree with the transaction of block which is sending by the old nodes.
- 3) The old nodes agree with the transaction of block which is sending by the new nodes.
- 4) The old nodes don’t agree with the transaction of block which is sending by the new nodes.

Because of these four different cases in getting consensus, fork problem happens, and according to these four cases, fork problems can be divided into two types, the Hard Fork and the Soft Fork. In addition to distinguish the new nodes and the old nodes, we have to compare the computing power of new nodes with old nodes, and assume that the computing power of new nodes are more than 50

- Hard Fork

Hard Fork means when system comes to a new version or new agreement, and it didn’t compatible with previous version, the old nodes couldn’t agree with the mining of new nodes, so one chain became two chains. Although new nodes computing power were stronger than old nodes, old nodes will still continue to maintain the chain which it though was right. Figure 5 shows the hard fork problem.

When Hard Fork happens, we have to request all nodes in the network to upgrade the agreement, the

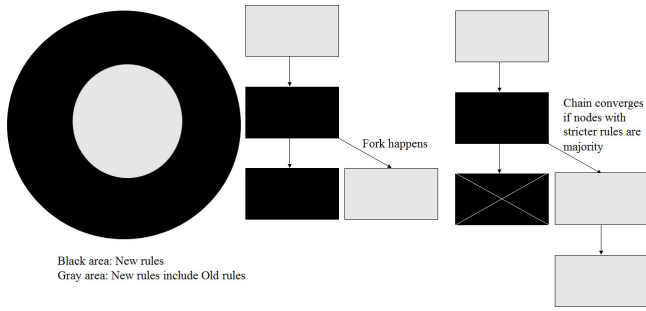


Figure 6: Hard Fork happens because the old node verification requirement is much stricter than the new node

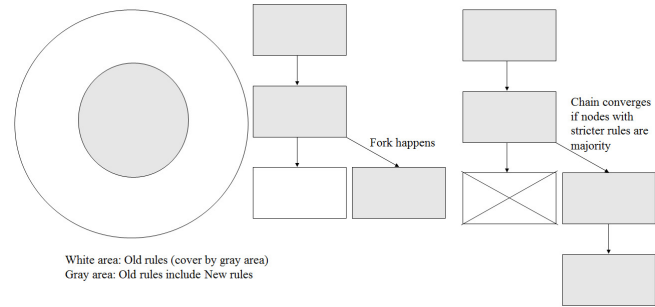


Figure 8: Soft Fork happens because the new node verification requirement is much stricter than the old node

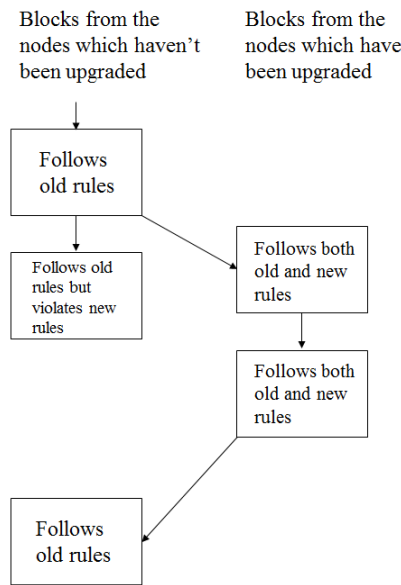


Figure 7: Compatible hard fork

nodes which haven't been upgrade will not continue to work as usual. If there were more old nodes didn't upgrade, then they will continue to work on the other completely different chain, which means the ordinary chain will fork into two chains. Figure 6 shows the reason of why hard fork will happens.

- Soft Fork

Soft Fork means when system comes to a new version or new agreement, and it didn't compatible with previous version, the new nodes couldn't agree with the mining of old nodes. Because the computing power of new nodes are stronger than old nodes, the block which is mining by the old nodes will never be approve by the new nodes, but new nodes and old nodes will still continue to work on the same chain. Figure 7 shows the soft fork problem.

When Soft Fork happens, nodes in the network don't have to upgrade the new agreement at the same time, it allows to upgrade gradually. Not like Hard Fork, Soft Fork will only have one chain, it won't affect the stability and effectiveness of system when nodes upgrade. However, Soft Fork makes the old nodes unaware that the consensus rule is changed, contrary to the principle of every nodes can verify correctly to some extent. Figure 8 shows the reason of why soft fork will happens.

4.3 Scale of Blockchain

As blockchain growing, data becomes bigger and bigger, the loading of store and computing will also getting harder and harder, it takes plenty of time to synchronize data, in the same time, data still continually increase, brings a big problem to client when running the system [10].

Simplified Payment Verification (SPV) is a payment verification technology, without maintain full blockchain information, only have to use block header message. This technology can greatly reduce user's storage in blockchain payment verification, lower the user's pressure when transaction drastically increased in the future.

4.4 Time Confirmation of Blockchain Data

Compared to traditional online credit card transaction, usually takes 2 or 3 days to confirm the transaction, bitcoin transaction only have to use about 1 hour to verify, it's much better than the usual, but it's still not good enough to what we want it to. **Lightning Network** is a solution to solve this problem [19].

Lightning Network is a proposed implementation of Hashed Timelock Contracts (HTLCs) with bi-directional payment channels which allows payments to be securely routed across multiple peer-to-peer payment channels. This allows the formation of a network where any peer

on the network can pay any other peer even if they don't directly have a channel open between each other.

4.5 Current Regulations Problems

Use Biction for example, the characteristics of decentralized system, will weak the central bank's ability to control the economic policy and the amount of money, that makes government be cautious of blockchain technologies, authorities have to research this new issue, accelerate formulating new policy, otherwise it will have risk on the market.

4.6 Integrated Cost Problem

Of course it will have lot of cost including time and money to change existing system, especially when it's an infrastructure. We have to make sure this innovative technology not only create economic benefits, meet the requirements of supervision, but also bridge with traditional organization, and it always encounter difficulties from internal organization which is existing now.

5 Conclusions

There's no doubt that blockchain is a hot issue in recent years, although it has some topics we need to notice, some problems has already been improved along with new technique's developing on application side, getting more and more mature and stable.

The government have to make corresponding laws for this technology, and enterprise should ready for embrace blockchain technologies, preventing it brings too much impact to current system.

When we enjoy in the advantage of blockchain technologies bring to us, in the same time, we still have to stay cautious on its influence and security issues that it could be have.

Acknowledgments

This study was supported by the National Science Council of Taiwan under grant NSC 105-2410-H-005 -023 -MY2. The authors gratefully acknowledge the anonymous reviewers for their valuable comments.

References

- [1] I. Bentov, A. Gabizon, and A. Mizrahi, "Cryptocurrencies without proof of work," *CoRR*, vol. abs/1406.5694, 2014.
- [2] J. Bonneau, A. Miller, J. Clark, A. Narayanan, J. A. Kroll, and E. W. Felten, "Sok: Research perspectives and challenges for bitcoin and cryptocurrencies," in *IEEE Symposium on Security and Privacy*, pp. 104–121, May 2015.
- [3] N. T. Courtois and L. Bahack, "On subversive miner strategies and block withholding attack in bitcoin digital currency," *CoRR*, vol. abs/1402.1718, 2014.
- [4] I. Eyal and E. G. Sirer, "Majority is not enough: Bitcoin mining is vulnerable," *CoRR*, vol. abs/1311.0243, 2013.
- [5] J. Garay, A. Kiayias, and N. Leonardos, *The Bitcoin Backbone Protocol: Analysis and Applications*, pp. 281–310, Springer Berlin Heidelberg, Berlin, Heidelberg, 2015.
- [6] A. Gervais, G. O. Karame, V. Capkun, and S. Capkun, "Is bitcoin a decentralized currency?," *IEEE Security Privacy*, vol. 12, pp. 54–60, May 2014.
- [7] A. Gervais, G. O. Karame, K. Wüst, V. Glykantzis, H. Ritzdorf, and S. Capkun, "On the security and performance of proof of work blockchains," in *Proceedings of ACM SIGSAC Conference on Computer and Communications Security (CCS'16)*, pp. 3–16, New York, NY, USA, 2016.
- [8] A. Gervais, H. Ritzdorf, G. O. Karame, and S. Capkun, "Tampering with the delivery of blocks and transactions in bitcoin," in *Proceedings of the 22Nd ACM SIGSAC Conference on Computer and Communications Security (CCS'15)*, pp. 692–705, New York, NY, USA, 2015.
- [9] E. Heilman, A. Kendler, A. Zohar, and S. Goldberg, "Eclipse attacks on bitcoin's peer-to-peer network," in *24th USENIX Security Symposium*, pp. 129–144, Washington, D.C., 2015.
- [10] G. Karame, "On the security and scalability of bitcoin's blockchain," in *Proceedings of ACM SIGSAC Conference on Computer and Communications Security (CCS'16)*, pp. 1861–1862, New York, NY, USA, 2016.
- [11] G. O. Karame, "Two bitcoins at the price of one? double-spending attacks on fast payments in bitcoin," in *Proceedings of Conference on Computer and Communication Security*, pp. 1–17, 2012.
- [12] S. King and S. Nadal, *Ppcoin: Peer-to-peer Crypto-Currency with Proof-of-Stake*, 2012. (https://archive.org/stream/PPCoinPaper/ppcoin-paper_djvu.txt)
- [13] A. Kosba, A. Miller, E. Shi, Z. Wen, and C. Papamanthou, "Hawk: The blockchain model of cryptography and privacy-preserving smart contracts," in *2016 IEEE Symposium on Security and Privacy (SP'16)*, pp. 839–858, May 2016.
- [14] L. Luu, V. Narayanan, C. Zheng, K. Baweja, S. Gilbert, and P. Saxena, "A secure sharding protocol for open blockchains," in *Proceedings of ACM SIGSAC Conference on Computer and Communications Security (CCS'16)*, pp. 17–30, New York, NY, USA, 2016.
- [15] S. Nakamoto, *Bitcoin: A Peer-to-Peer Electronic Cash System*, Feb. 24, 2013. (<http://bitcoin.org/bitcoin.pdf>)
- [16] E. U. Opara, O. A. Soluade, "Straddling the next cyber frontier: The empirical analysis on network

security, exploits, and vulnerabilities,” *International Journal of Electronics and Information Engineering*, vol. 3, no. 1, pp. 10–18, 2015.

- [17] M. Rosenfeld, “Analysis of hashrate-based double spending,” *CoRR*, vol. abs/1402.2009, 2014.
- [18] J. Singh, “Cyber-attacks in cloud computing: A case study,” *International Journal of Electronics and Information Engineering*, vol. 1, no. 2, pp. 78–87, 2014.
- [19] Y. Sompolinsky and A. Zohar, *Secure High-Rate Transaction Processing in Bitcoin*, pp. 507–527, Springer Berlin Heidelberg, Berlin, Heidelberg, 2015.
- [20] W. T. Tsai, R. Blower, Y. Zhu, and L. Yu, “A system view of financial blockchains,” in *IEEE Symposium on Service-Oriented System Engineering (SOSE’16)*, pp. 450–457, Mar. 2016.
- [21] H. Watanabe, S. Fujimura, A. Nakadaira, Y. Miyazaki, A. Akutsu, and J. Kishigami, “Blockchain contract: Securing a blockchain applied to smart contracts,” in *IEEE International Conference on Consumer Electronics (ICCE’16)*, pp. 467–468, Jan. 2016.

Biography

Iuon-Chang Lin received the B.S. in Computer and Information Sciences from Tung Hai University, Taichung, Taiwan, Republic of China, in 1998; the M.S. in Information Management from Chaoyang University of Technology, Taiwan, in 2000. He received his Ph.D. in Computer Science and Information Engineering in March 2004 from National Chung Cheng University, Chiayi, Taiwan. He is currently a professor of the Department of Management Information Systems, National Chung Hsing University, and Department of Photonics and Communication Engineering, Asia University, Taichung, Taiwan, ROC. His current research interests include electronic commerce, information security, cryptography, and mobile communications.

Tzu-Chun Liao graduated from National Chung Hsing University, Taichung, Taiwan, Republic of China, in 2015; He is currently an M.S. student of the Department of Management Information Systems, National Chung Hsing University.

Efficient Feature Selection Technique for Network Intrusion Detection System Using Discrete Differential Evolution and Decision Tree

Ebenezer Popoola¹ and Aderemi Adewumi²

(Corresponding author: Aderemi Adewumi)

School of Mathematics, Statistics & Computer Science, University of KwaZulu-Natal¹
Durban, 4000, South Africa

Email: Adewumia@ukzn.ac.za & popoolaebenezer@gmail.com

(Received June 30, 2016; revised and accepted Sept. 3 & Oct. 11, 2016)

Abstract

Network intrusion is a critical challenge in information and communication systems amongst other forms of fraud perpetrated over the Internet. Despite the various traditional techniques proposed to prevent this intrusion, the threat persists. These days, intrusion detection systems (IDS) are faced with detecting attacks in large streams of connections due to the sporadic increase in network traffics. Although machine learning (ML) has been introduced in IDS to deal with finding patterns in big data, the irrelevant features in the data tend to degrade both the speed and accuracy of detection of attacks. Also, it increases the computational resource needed during training and testing of IDS models. Therefore, in this paper, we seek to find the optimal feature set using discretized differential evolution (DDE) and C4.5 ML algorithm from NSL-KDD standard intrusion dataset. The result obtained shows a significant improvement in detection accuracy, a reduction in training and testing time using the reduced feature set. The method also buttresses the fact that differential evolution (DE) is not limited to optimization of continuous problems but work well for discrete optimization.

Keywords: C4.5; Differential Evolution; Machine Learning; Network Intrusion Detection; NSL-KDD

1 Introduction

Asides other challenges faced by computer network systems, network intrusion remains one of the unresolved issues. It has drawn the attention of researchers due to its continuous threat to the sustenance of businesses. Intrusion is described as any form of activity which tends to breach the availability, confidentiality or security of networks [13, 16]. The goal of IDS is to identify normal connections from both those using computer resources with-

out authorization or abusing the privileges given to them. However, IDS are faced with a greater challenge as the network traffic grows [3, 21].

According to [4] over 25 billion devices will be connected by 2020 which will lead to more traffic as services are moved to the cloud due to ease of management and reduced running cost. Therefore, to sustain this technological advancement, there is a need for swift action in solving this network threat. IDS is categorized into two namely “Misuse” and “Anomaly” Detection [24]. While misuse detection deals with identifying known patterns of intrusion by comparing it with previous signatures in the database, anomaly detection identifies patterns that deviate from the standard pattern. This work focuses on anomaly detection because attackers tend to change their method of intrusion when discovered thereby making misuse detectors useless.

To detect anomaly in networks, several methods have been proposed among which ML based IDS have shown to be more efficient. Despite its advantage over other methods, it still suffers from high false positive rate (FPR), false negative rate (FNR), expensive computational cost and slow classification during training and testing. These downsides defeat the purpose of IDS since it is expected to be fast and accurate. Many works attribute this deficiency to irrelevant features in the dataset and have proposed various feature reduction techniques as the solution. Some nature inspired (NI) algorithms have shown to be effective in searching for an optimal set of features. These include Genetic Algorithm (GA) [26], Particle Swarm Optimization (PSO) [17], Ant Colony Optimization (ACO) [33], etc. but little has been done using Differential Evolution (DE). Hence, this work hybridizes DDE and C4.5 ML algorithm for finding an optimal set of feature.

2 Related Works

To address the issue of network intrusion, various steps has been taken. Some work focused on finding the best parameter settings for the classifier while others addressed removing irrelevant features from the dataset.

Due to unavailability of a standard labelled dataset, most research on IDS has used DARPA dataset known as KDDCUP'99 [19]. But investigation shows that the dataset has some fundamental issue which places a shadow of doubt on the usefulness of models designed with it [18]. Some of the problems are:

- 1) Redundant records in the dataset which causes the classifier to see less frequent records as noise. This leads to misclassification.
- 2) Duplication of records which made some models have better detection rates on the reoccurring data.
- 3) The above leads to having too large dataset which leads to excessive computational time. Therefore, most works randomly select a subset of the dataset which gives no basis to compare different models or performance of different feature set. Also, this leaves no basis for comparing the speed of IDS.

NSL-KDD [20] provided by [32] is a refined subset of the KDDCUP'99 dataset which solves the issues listed above. It was validated by testing it on some ML algorithms using their default parameters without a reduction in feature set. This serves as a benchmark to subsequent models. For binary classification ("Normal" or "Attack"), the following accuracies were obtained from the dataset.

This refined dataset has been validated using 7 ML algorithms on WEKA tool [14] without tuning the parameters of the learners. The resulting classification accuracy per classifier are as follows; NB-Tree: 82.67%, J48: 81.05%, Random Forest (RF): 81.59%, Multi-Layer Perceptron (MLP): 77.41%, Naive Bayes (NB): 76.56% and Support vector machine (SVM): 69.52%.

Using this dataset, a various number approach is being taken to either increase the classification accuracy or reduce the time needed during training and testing IDS. In some case, both objectives are achieved. One approach is to find optimal parameter setting of the classifier, and another is to tactically reduce the feature used to achieve faster training and testing time.

Garg and Kumar [13] reviewed various selection and classification techniques. The work tested the performance of combining two to three feature selection methods using Boolean AND operation. Out of 10 techniques tested, the combination of Symmetric and Gain Ratio for feature selection using 15 features and IBK classifier yielded the highest accuracy. However, no reason was given on why and how random data was selected from the dataset. Hence, the result can not be replicated.

Aziz et al. [5] also compared the performance of correlation-based feature selection (CFS), sequential floating selection (SFS), principal component analysis

(PCA), information gain and rough sets in selecting appropriate features. SFS methods performed best using 26 features.

Al-Jarrah et al. [2] proposed two novel feature selection techniques: RandomForest-Forward Selection Ranking (RF-FSR) and RandomForest-Backward Elimination Ranking (RF-BER). Although the 15 features selected using RF-FSR achieved higher classification accuracy, it was only tested using cross-validation which does not guarantee its usability as reflected in [22].

Gaikwad and Thool [12] proposed Bagging ensemble classifier method on NSL-KDD dataset using the Rep-Tree algorithm as the base class. The method utilized a reduced feature set of 29 features to achieve an accuracy of 81.2988% on the test set and 99.6761% using 10-fold cross-validation on the training set. Also, [27] compared the performance of selected ML algorithms where Bagging ensemble selection algorithm performed best with 97.85% accuracy after cross-validation. Ingre and Yadav [15] performed an analysis of NSL-KDD dataset using Artificial Neural Network (ANN) for both binary class classification (normal and attack) and five class classification (normal, DoS, U2R, R2L, and Probe). The method achieved 81.2% and 79.9% accuracies showing binary classification performs better than multiclass. Pervez and Farid [22] proposed an SVM-based feature selection method which achieved 91% accuracy using three features and 99% with 41 features on all the training set; the setback is that its classification accuracy when tested with an unseen dataset give 82.37%. Deshmukh et al. [10] approached intrusion detection by normalizing, discretized and selecting feature before training three classifiers (NB-Tree, NB, and AD Tree). The output showed NB algorithm performed efficiently regarding effectiveness, elegance, simplicity and robustness when compared to others on the training set. Since [10] did not provide this result when further evaluated on the test set, it is uncertain whether the model would yield high detection rate as results from previously proposed methods show that a training model may give a high accuracy, but does not guarantee the same when tested with new attacks.

Lately, nature-inspired (NI) algorithm was introduced to solving either feature selection or parameter optimization. Benaicha et al. [6], considered GA in detecting some attacks in NSL-KDD dataset using nine selected features, the focus was on detecting some DoS attacks (Neptune, smurf, teardrop, pod, back). The result was quite impressive but did not address other DoS attacks in test data and other attacks in NSL-KDD.

Likewise, [7] proposed an IDS which used GA, information gain (IG), mutual correlation and cardinality of features to achieve higher accuracy. The IG based feature selection was able to increase the accuracy to 87.54% using nine features. It was also limited to detecting Neptune, Satan, and Smurf attacks which are under the denial of service (DoS). In [5] network intrusion was approached by reviewing GA with different feature selection methods. Correlation-based feature selection techniques such

as IG, sequential floating, rough set and principal component analysis (PCA) for feature extraction were used on NSL-KDD dataset. Sequential-floating backward selection showed to be more effective with higher detection rates.

Stein et al. [28] proposed a model which considered decision tree classifier for network intrusion detection with GA-based feature selection. Their work was focused on identifying features which could separate each category of attacks (DoS, Probe, U2R, and R2L) from normal connections in a network. This led to creating four different models which were yet to be integrated as one.

Despite the superiority of DE in solving various optimization [9], there is a limited usage in its application to intrusion detection. Elsayed et al. [11] proposed DE for classification of attack but applied flexible neural tree (FNT) [8] for feature selection. Here, we propose a hybridization of DDE and C4.5 for selecting optimal features.

3 Problem Definition

Though some IDS adopts the working principle of ML for classification of connections in a network, it still faces a critical challenge which limits its use in some real life environment. Some systems proposed are either slow or raises false alarms, e.g., misclassification of normal connection as intrusive which could frustrate the experience of client or classification of intrusive connection as normal leading to a significant loss. Also, the fundamental issues about KDDCUP'99 dataset [18] give doubt on the accuracy presented by most works. For an IDS to be effective, the classification accuracy must be high, and detection rate must be fast without the usage of excessive computational resources. To achieve this goal, there is the need for removing irrelevant features from network connections during classification. Hence, the major problem is how to select the best set of feature for the right ML algorithm without loss of relevant information. Several feature reduction methods have been proposed including the use of other computational intelligence techniques such fuzzy systems, neural networks (NN) and NI algorithms. Among the various NI algorithms, GA show to be the favorite optimization technical several works has been done while little or no work has been done using DE for finding features for IDS.

Generally, GA concentrates more on exploring the search space than exploiting the environment of a solution. It is however noted that exploiting a weak solution using DE could yield a better solution than the best in the population for discrete optimization problems which in turn reduces the number of generations. Hence, we utilize DE for finding the optimal set of features where the selection criteria are based on the classification accuracy of the feature set. We also emphasize the need for using NSL-KDD dataset [32] during modelling which solves the fundamental issues of the KDDCUP'99 dataset.

4 Methodology

This section explains various concepts such as C4.5 Decision Trees (DT), DE and how they are harnessed in our proposed technique to search for the optimal feature set.

4.1 Decision Trees (C4.5)

DT designed by [23] are flow-chart-like structures which follow an IF-THEN rule have proved to be fast and efficient. It is an improvement of the ID3 algorithm which is a top-down recursive divide and conquer approach. It seeks to divide the entire dataset by first breaking the values of a feature(or Attribute) into ranges. To do this efficiently at each node, C4.5 calculates the gain ratio of each attribute and selects the one with highest the gain ratio. The expected information, Entropy, Information gain and gain ratio are calculated to ensure the best split is achieved which can be seen as follows.

Let S be the sample data to be split which consist of X_1, \dots, X_j attributes where the j^{th} represents the distinct class labels(C_i, \dots, C_m)

$$H(S) = - \sum_{i=1}^m P_i \log_2(P_i), \quad (1)$$

where, H is the expected information that is needed to classify a given sample data. P_i is the probability that an instance in sample S belongs to class C_i which is calculated as the sum of all instances in S which are of class C_i divided by the total instance in S ($|C_{i,s}|/|S|$).

Assuming the range of values of X_1 is subdivided into v distinct values $\{x_1, x_2, \dots, x_v\}$ and used to divide S into $\{S_1, S_2, \dots, S_v\}$ where S_j contains instances in D that corresponds to x_i of X_1 . Hence the entropy which is the expected information needed to classify an instance from S based on splitting by X_1 in order to arrive at an exact classification can be given as:

$$info_{X_1}(S) = \sum_{j=1}^v \frac{|S_j|}{|S|} \times H(S_j). \quad (2)$$

It is noted that $\frac{|S_j|}{|S|}$ is the weight of the J^{th} partition and the smaller the value of $info_{X_1}(S)$, the greater the possibility that all instances in the partition belong to the same class.

Now, information gain is the expected reduction in entropy caused by partitioning the samples according to the attribute X_1 . That is, the difference between splitting S based on the proportion of classes and partitioning on X_1 which is given as:

$$Gain(X_1) = H(S) - info_{X_1}(S). \quad (3)$$

Gain Ratio is used to reduce a bias towards multi-valued attributes by taking the number and size of branches into account when choosing an attribute.

$$GainRatio(X_1) = Gain(X_1)/SplitInfo(X_1). \quad (4)$$

Some other advantages of C4.5 can be extracted from [1]. Various DT classifiers including C4.5 adopts a greedy approach as it considers the immediate consequence of selecting a feature to split on at the current node without estimating the full depth future implication. Hence, a need to devise a method such that features with a high gain ratio at the initial split but more misclassification cost at the end is removed.

4.2 Differential Evolution

DE [30] which is also a population-based search technique as GA is the main optimization technique explored in this work. The sequence of operating is given as initialization, mutation, recombination/crossover and selection [25]. Although DE was originally designed for continuous problems, we see how to adapt its concept on the discrete problem through representing the solutions like that of GA. Also, using similar operators of GA but in the sequence of DE as shown in Figure 1. The details are as follows.

- **Initialization:** An initial population of chromosomes is generated for the first generation: $X_G = \{x_{1,G}, x_{2,G}, x_{3,G}, \dots, x_{i,G}\}$ where each chromosome $x_{i,G}$ is called the **target vector**.
- **Mutation:** The mutation is different from that of GA. Here, for each $x_{i,G}$, a **donor vector** $v_{i,G+1}$ is generated.

$$v_{i,G+1} = x_{r1,G} + F.(x_{r2,G} - x_{r3,G}), \tag{5}$$

where $x_{r1,G}, x_{r2,G}, x_{r3,G}$ are randomly chosen from the population excluding the target vector. F is a user defined **mutation or constant factor** ($F \in [0, 2]$) which controls the amplification of the differential variation ($x_{r2,G} - x_{r3,G}$).

- **Crossover:** The mutant vector is mixed with the target vector to produce a **trial vector** $u_{i,G+1}$ as follows:

$$u_{i,G+1} = \begin{cases} v_{j,i,G+1} & \text{if } rand_{j,i} \leq CR \text{ or } j = I_{rand} \\ x_{j,i,G} & \text{if } rand_{j,i} > CR \text{ and } j \neq I_{rand} \end{cases}$$

$i = 1, 2, \dots, N; j = 1, 2, \dots, D$, where N is the population size and D is the dimension of x_i . CR is the crossover constant ($[0, 1]$). I_{rand} ensures $v_{i,G+1} \neq x_{i,G}$.

- **Selection:** The trial vector $u_{i,G+1}$ and target vector $x_{i,G}$ are evaluated using the objective function. A comparison is then made between the two. The one with lower value is moved to the new generation.

$$x_{i,G+1} = \begin{cases} u_{i,G+1} & \text{if } f(u_{i,G+1}) \leq f(x_{i,G}) \\ x_{i,G} & \text{otherwise} \end{cases}$$

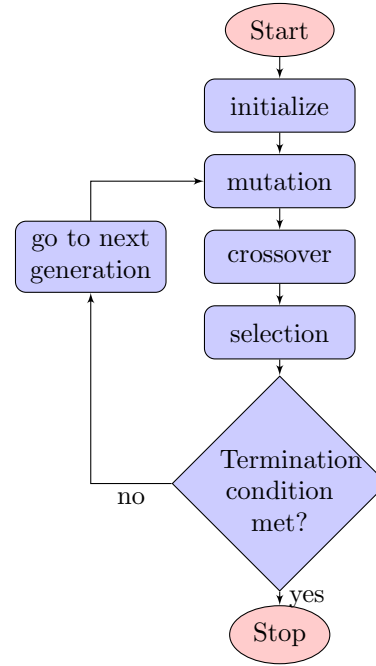


Figure 1: Differential evolution

4.3 Proposed Method

The proposed method models a DDE [31] strategy for finding the optimal set of features as follows.

4.3.1 Initialization of Population

Since the optimization problem deals with whether or not a feature is used, the solutions are expected are binary strings of 1 and 0 where a gene position having 0 means the feature is not used. However for the DE operators to have more effect, we initially encode the 41 positions as random numbers in the range of 0 and 1. Table 1 gives a picture of how a solution is represented.

Table 1: Chromosome

0.5	0.1	0.9	0.7	0.55	0.41	0.1	0.3
-----	-----	-----	-----	------	------	-----	-----

4.3.2 Mutation

To generate the mutant vector v_i for a target vector $x_{i,G}$, we do the following:

$$v_{i,G+1} = m \bigoplus F.(x_{r,G}) \tag{6}$$

where: $m = \{a, a, a, \dots \mid |m| = |x_{r,G}|, 0 < a < 1\}$ in this case, $a = 0.3$ and $F = 0.4$ which is a scalar value.

$x_{r,G}$ is a random chromosome such that $x_{i,G} \neq x_{r,G}$. While F is the amplification factor which is preset, m gives some stability since the element of $x_{r,G}$ are random numbers in the range of 0 and 1.

4.3.3 Crossover

The mutant vector is mixed with the target vector to produce a **trial vector** $u_{i,G+1}$ as follows:

$$u_{i,G+1} = \begin{cases} v_{j,i,G+1} & \text{if } rand_{j,i} \leq CR \text{ or } j = I_{rand} \\ x_{j,i,G} & \text{if } rand_{j,i} > CR \text{ and } j \neq I_{rand} \end{cases}$$

$i = 1, 2, \dots, N; j = 1, 2, \dots, D$, where $N = 10$ is the population size and D is the dimension of x_i . CR is the crossover constant ($[0, 1]$) which is set at 0.5.

4.3.4 Selection

The following test is done in parallel with fitness evaluation in the next section to select the candidate for the next generation.

$$x_{i,G+1} = \begin{cases} x_{i,G} & \text{if } u_{i,G+1} = \{0, 0, 0, \dots\} \text{ or } \{1, 1, 1, \dots\} \\ u_{i,G+1} & \text{if } f(u_{i,G+1}) \leq f(x_{i,G}) \\ x_{i,G} & \text{otherwise} \end{cases}$$

If the first condition is met, it leads to an outright reject of the trial vector because it means none or all the features are selected but the goal is to find optimal features. The second selects the trial vector if the fitness is less than the target vector since we are minimizing the misclassification rate.

4.3.5 Fitness Evaluation

The fitness of each chromosome is evaluated by converting the genes to 0's and 1's based threshold values and mapping the gene position to the features in the dataset. Taking note that every gene whose value is 0 indicates the feature is not used for classification while the reverse is the case where the value is 1. This reduces the dataset column-wise before been passed to WEKA tool where classification is done using the C4.5 algorithm. The accuracy after training and testing becomes the fitness value which is calculated as follows:

$$accuracy = \frac{\text{correctly classified instance}}{\text{total test set}}$$

$$\text{misclassification rate} = 1 - \text{accuracy}.$$

4.3.6 Algorithm for Feature Selection

The algorithm 1 shows how features were selected.

4.3.7 Performance Metric Used For The Proposed Method

There are standard performance metrics used in evaluating of an ML algorithm which can either be done through the command line or graphically using the WEKA tool. In this work, the following metrics are verified after obtaining the optimal feature set from the search technique:

Algorithm 1 Feature selection using discrete DE

```

1: Initialize Population
2: Initialize  $m$ ,  $F$ ,  $CR$ , Generation  $G$  and conversion threshold  $tr$ 
3: for  $i \leftarrow 1$  to  $G$  do
4:   for each chromosome do
5:     Set target vector  $(x_{i,G}) =$  chromosome
6:     Randomly select  $x_{r,G} || x_{r,G} \neq x_{i,G}$ 
7:     Mutate to get the trial mutant vector  $v_{i,G+1}$ 
8:     crossover each allele of  $v_{i,G+1}$  and  $x_{i,G}$  with a Probability of CR to get  $u_{i,G+1}$ 
9:     Selection
10:    if All element of  $u_{i,G+1}$  0's or 1's then
11:      Add target vector to new population
12:    else
13:      Convert alleles to 0 or 1 based  $tr$ 
14:      Evaluate  $f(u_{i,G+1})$  and  $(x_{i,G})$ 
15:      if  $f(u_{i,G+1}) < f(x_{i,G})$  then
16:        Add initial trial vector before conversion to new population
17:      else
18:        Add initial target vector before conversion to new population
19:      end if
20:    end if
21:  end for
22: end for
23: map the genes of the chromosome to its actual feature name

```

- **Sensitivity or True positive rate (TPR):** This is the ratio of positive class (attack connections) that are correctly identified to the total positive class (P).

$$\text{True positive rate} = \frac{\text{True positives (TP)}}{P}.$$

- **True negative rate (TNR) or specificity:** It estimates the ratio of negative class (normal connections) that are correctly identified to the total negative class (N).

$$\text{True negative rate} = \frac{\text{True negative (TN)}}{N}.$$

- **Precision** is the measure of exactness, i.e., the percentage of instances classified as an attack, out of the total number of cases classified as attacks.

$$\text{Precision} = \frac{TP}{TP + FP},$$

where: FP (**false positives**) are normal which are wrongly classified as attacks while the counterpart called **false negatives** (FN) are attack connections wrongly classified as normal.

- **Recall** shows the measure of completeness. It is calculated just like true positive rate.

- **error or misclassification rate** is calculated as $1 - \text{accuracy}$. It can also be computed as

$$\text{misclassification rate} = \frac{FP + FN}{P + N}.$$

- **Receiver Operating Characteristics (ROC) Curve** is a plot that shows the trade-off between TPR and the rate of false positives (FPR). It is a visual representation of the rate at which the proposed model can recognize normal connections versus misclassification of attacks as normal for various sections of the dataset. Also, it leaves an **area under the curve (AUC)** which also determines how well the classifier performs. The closer the area is to 1.0, the better the classifier.

5 Experimental Setting

The implementation of the proposed method was written mainly with Python 2.7 programming language and an ML library known as WEKA, version 3.8.0 which has several learning algorithms including C4.5. With respect to the hardware used, a personal computer having a random access memory of 4GB RAM, storage size 500GB with Processor type Intel(R) Core(TM) i3 CPU @2.53GHz speed was used. Since WEKA is written in Java, its integration with python is made possible using java virtual machine wrapper for Python.

6 Dataset

The KDDCup'99 dataset contains historical data prepared by [29] for evaluating IDSs. It has been a common dataset used for training and testing models of ML algorithms, hence used as a benchmark. The dataset comprises of a training and test set. The Training set contains 22 different types of attacks which are mixed with normal connections while the test set contains both the 22 attacks and 17 news attack which total to 39 attacks. The records in the dataset consist of 41 attributes (features). Due to the inherent challenges of the dataset, NSL-KDD dataset which is an extract from the original KDDCUP'99 dataset is used. It contains about 125973 instances of TCP/IP connections which can be utilized for training and 22544 instances for testing designed models. The instances in the dataset are data connections as they flow from source to destination. For experimental purpose, each instance is given a "label" to identify it as either an "attack" or "normal" connection. The values of the features can also be divided based on their type: numeric or symbolic. One of the advantages of NSL-KDD dataset is that the quantity is reasonable enough to be trained as a whole by most algorithm as compared to KDDCUP'99 where small portions of the dataset are used for training and testing [32]. Table 2 shows the list of features and their types.

Also, both the 22 attacks in the training set and 39 attacks in the test set can be further categorized into four

Table 2: Total list of features in NSL-KDD dataset

No.	Feature	Type
1	Duration	Numeric
2	protocol_type	Symbolic
3	Service	Symbolic
4	flag	Symbolic
5	src_bytes	Numeric
6	dst_bytes	Numeric
7	land	Numeric
8	wrong_fragment	Numeric
9	urgent	Numeric
10	hot	Numeric
11	num_failed_logins	Numeric
12	logged_in	Numeric
13	num_compromised	Numeric
14	root_shell	Numeric
15	su_attempted	Numeric
16	num_root	Numeric
17	num_file_creations	Numeric
18	num_shells	Numeric
19	num_access_files	Numeric
20	num_outbound_cmds	Numeric
21	is_host_login	Numeric
22	is_guest_login	Numeric
23	count	Numeric
24	srv_count	Numeric
25	serror_rate	Numeric
26	srv_serror_rate	Numeric
27	rerror_rate	Numeric
28	srv_rerror_rate	Numeric
29	same_srv_rate	Numeric
30	diff_srv_rate	Numeric
31	srv_diff_host_rate	Numeric
32	dst_host_count	Numeric
33	dst_host_srv_count	Numeric
34	dst_host_same_srv_rate	Numeric
35	dst_host_diff_srv_rate	Numeric
36	dst_host_same_src_port_rate	Numeric
37	dst_host_srv_diff_host_rate	Numeric
38	dst_host_serror_rate	Numeric

39	dst_host_srv_serror_rate	Numeric
40	dst_host_error_rate	Numeric
41	dst_host_srv_rerror_rate	Numeric

broader groups which are Dos, Probe, U2R and R2L as described below also in Tables 3 and 4:

- **Denial of Service(DoS)**: An attack which creates excessive computational request to the responding system such that there are no more resources on the destination side to respond to a request from legitimate users which is the goal of the attacker.
- **Probe attacks**: The attacker gets sensitive information about a network by scanning data without access permission with the aim of breaking its security control.
- **Remote-to-Local (R2L)**: uses the vulnerability of a system to get a normal user account from a remote location to gain local access to a system.
- **User-to-Root(U2R)**: It is when an attacker uses means such as social engineering, password sniffing to get the password of a normal user of a system, from there he can exploit some vulnerability of the system to gain root access.

Table 3: Categories of attacks in NSL-KDD training and test dataset

Category	Actual Attacks in Training	Additional Attacks in Test set
DoS	Neptune, smurf, teardrop, pod, land, back	apache2, mailbomb, processtable, udp-storm
Probing	satan, ipsweep, nmap, portsweep	mscan, saint
R2L	imap, warezmaster, phf, multihop, guess_passwd, spy, warezclient, ftp_write	httptunnel, named, sendmail, snmpgetattack, xlock, xsnoop
U2R	loadmodule, buffer_overflow, rootkit, perl	ps, snmpguess, sqlattack, worm, xterm

7 Results and Discussion

This section provides experimental results of the proposed feature selection search technique. It uses the standard

Table 4: Categories of attacks in NSL-KDD training and test dataset

Training Dataset		Testing dataset	
Attack Class	Quantity	Attack Class	Quantity
Normal	67343	Normal	9711
DoS	54927	DoS	7458
Probe	11656	Probe	2421
R2L	995	R2L	2754
U2R	52	U2R	200
Total	125973	Total	22544

ML metric to evaluate the strength of the resulting classifier. These metric includes classification rate, FPR, TPR, precision, recall, F-measure and Auc. Also, it is compared to recent ML techniques to see how well it performs.

The performance of the classifier is evaluated both on the training set and more importantly the test set provided by NSL-KDD. Table 5 shows a detailed performance of the proposed feature set on the C4.5 algorithm. Comparing the result of Table 5 with Table 6 where Bagging ensembles was proposed by [12], it is clear that our method performs better both regarding accuracy and FPR. Also, Table 8 presents results from [22] which indicate the proposed model is better when matched. Likewise, It performs better in terms of accuracy than both multi and binary classification using ANN proposed by [15] as shown in Table 7.

Furthermore, the area under the curve (AUC) after feature selection in Figure 3 shows an appreciable increase (Auc = 0.9172) compared to Figure 2 (Auc = 0.84) which shows the classifier is balanced in its classification.

Table 5: Performance of proposed model on NSL-KDD dataset

Datasets	Training (100%)	Testing
Classification rate	99.81%	88.73%
Error rate	0.1873%	11.27%
TP Rate (Weighted Avg.)	0.998	0.89
FP Rate (Weighted Avg.)	0.002	0.093
Precision (Weighted Avg.)	0.998	0.90
Recall (Weighted Avg.)	0.998	0.89
F-Measure (Weighted Avg.)	0.998	0.89

Table 6: Performance on NSL-KDD dataset [12]

Performance Metric	On Training Set	On Test Set
Classifier Accuracy	99.6761%	81.2988%
False Positive Rate	0.003	0.148

Table 7: Performance on NSL-KDD dataset [15]

Classifier Method	Detection Accuracy	FPR (attack)
SOM	75.49	5.77
binary	81.2	4.23
five class	79.9	-

Table 8: Performance on NSL-KDD dataset [22]

Datasets	Training (100%)	Testing
Classification rate	99.01%	82.37%
Error rate	0.98%	17.62%
TP Rate (Weighted Avg.)	0.99	0.82
FP Rate (Weighted Avg.)	0.007	0.15
Precision (Weighted Avg.)	0.99	0.74
Recall (Weighted Avg.)	0.99	0.82
F-Measure (Weighted Avg.)	0.99	0.77

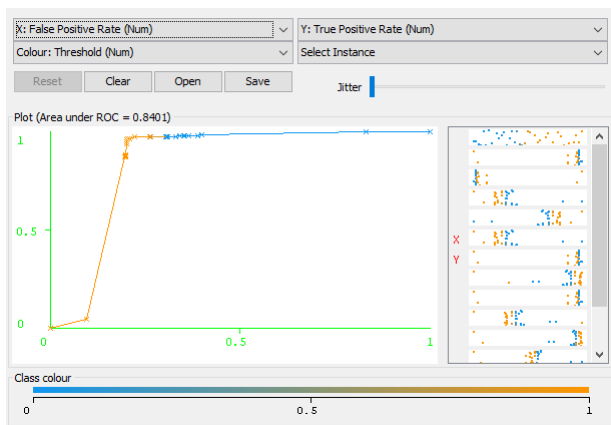


Figure 2: ROC before feature selection

We can also see the effect of the feature reduction in the training time of C4.5, as shown in Figure 4, which shows a 76% decrease in the training time.

In general, the achievements of this classifier is attributed to the choice of a feature set. The DDE algorithm helped in obtaining the right set of features as

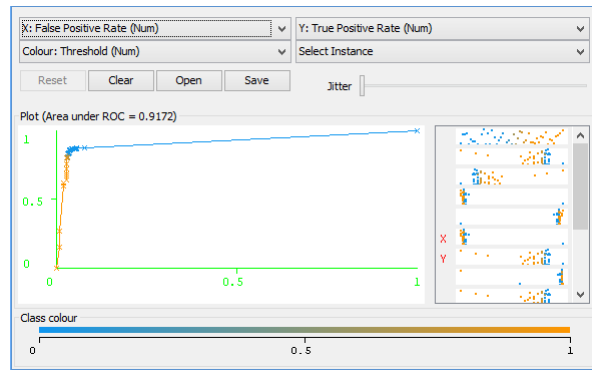


Figure 3: ROC after feature selection

Table 9: List of features used

No.	Feature
1	Duration
2	Service
3	src.bytes
4	land
5	hot
6	num_compromised
7	root_shell
8	is_guest_login
9	error_rate
10	rerror_rate
11	srv_error_rate
12	dst_host_count
13	dst_host_srv_count
14	dst_host_same_srv_rate
15	dst_host_srv_error_rate
16	dst_host_rerror_rate

shown in Table 9. It was observed that the feature used by C4.5 as the root node of the tree, before and after the proposed feature selection technique was “src.bytes” which tags the feature as the most important for their initial split. On the other hand, the “service” feature contributed most in making the final split. Also, despite the reduction of features, the important basic TCP features (1 - 4), content-based features (4 - 8), time-related traffic features (9 - 11), and host-based traffic features (12 - 16) were captured without loss in accuracy.

8 Conclusion

This work proposed an efficient feature selection technique for NID using DDE. Evaluation of the proposed

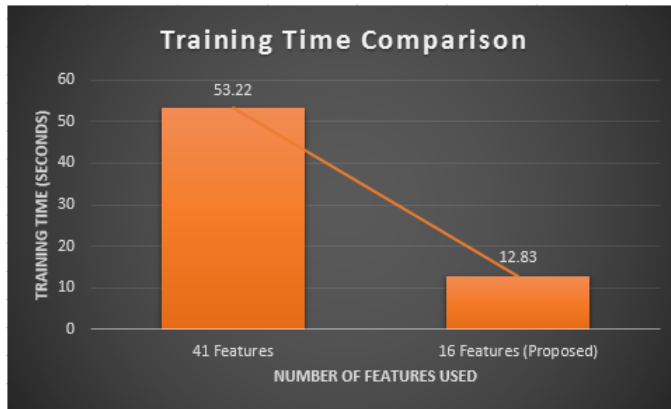


Figure 4: Comparison of 41 vs 16 features training times

method is done to both on the training and test set to rate the resulting classifier compared to other existing classifiers using standard ML performance metric. The computational result shows that this technique is able to identify 16 features capable of classifying the connections in the NSL-KDD dataset with high accuracy, low error and FPR. While it achieves 99.92% classification accuracy on the training set using 10-fold cross-validation, it is able to classify new attacks from in the test set with 88.73% accuracy. Aside the result above, the reduced feature set helps in reducing both the training and testing time used by the classifier (C4.5). Hence, the proposed method is greatly encouraged.

In future, we intend to extract connections from live networks having these set of features but with recent forms of attack to further test the model, because other datasets which have been provided by some authors do not have the exact features as in the NSL-KDD dataset.

Acknowledgments

The University of KwaZulu-Natal, South Africa supported this study. The authors gratefully acknowledge the anonymous reviewers for their valuable comments.

References

- [1] G. L. Agrawal, H. Gupta, "Optimization of C4. 5 decision tree algorithm for data mining application," *International Journal of Emerging Technology and Advanced Engineering*, vol. 3, no. 3, pp. 341–345, 2013.
- [2] O. Y. Al-Jarrah, A. E. M. Siddiqui, P. D. Yoo, S. Muhaidat, K. Kim, "Machine-learning-based feature selection techniques for large-scale network intrusion detection," in *Proceedings of The 34th International Conference on Distributed Computing Systems Workshops (ICDCSW'14)*, pp. 177–181, Madrid, Spain, June-July 2014.
- [3] A. Anurag, "Network neutrality: Developing business model and evidence based net neutrality regulation," *International Journal of Electronics and Information Engineering*, vol. 3, no. 1, pp. 1–9, 2015.
- [4] L. Atzori, A. Iera, G. Morabito, "The internet of things: A survey," *Computer Networks*, vol. 54, no. 15, pp. 2787–2805, 2010.
- [5] A. S. A. Aziz, A. T. Azar, M. A. Salama, A. E. Hassanien, S. E. O. Hanafy, "Genetic algorithm with different feature selection techniques for anomaly detectors generation," in *Proceedings of The Federated Conference on Computer Science and Information Systems (FedCSIS'13)*, pp. 769–774, Krakow, Poland, Sept. 2013.
- [6] S. E. Benaicha, L. Saoudi, S. E. B. Guermeche, O. Lounis, "Intrusion detection system using genetic algorithm," in *Proceedings of The Conference on Science and Information Conference (SAI'14)*, pp. 564–568, London, UK, Aug. 2014.
- [7] N. Cleetus, K. A. Dhanya, "Genetic algorithm with different feature selection method for intrusion detection," in *Proceedings of The First International Conference on Computational Systems and Communications (ICCSC'14)*, pp. 220–225, Dec. 2014.
- [8] Y. Chen, A. Abraham, B. Yang, "Feature selection and classification using flexible neural tree," *Neurocomputing*, vol. 70, no. 1, pp. 305–313, 2006.
- [9] S. Das, P. N. Suganthan, "Differential evolution: A survey of the state-of-the-art," *IEEE Transactions on Evolutionary Computation*, vol. 15, no. 1, pp. 4–31, 2011.
- [10] D. H. Deshmukh, T. Ghorpade, P. Padiya, "Intrusion detection system by improved preprocessing methods and naïve bayes classifier using NSL-KDD 99 dataset," in *Proceedings of The International Conference on Electronics and Communication Systems (ICECS'14)*, pp. 1–7, Feb. 2014.
- [11] S. Elsayed, R. Sarker, J. Slay, "Evaluating the performance of a differential evolution algorithm in anomaly detection," in *Proceedings of The Congress on Evolutionary Computation (CEC'15)*, pp. 2490–2497, Sendai, Japan, May 2015.
- [12] D. P. Gaikwad, R. C. Thool, "Intrusion detection system using bagging ensemble method of machine learning," in *Proceedings of The International Conference on Computing Communication Control and Automation (ICCCUBEA '15)*, pp. 291–295, Pune, Maharashtra, India, Feb. 2015.
- [13] T. Garg, Y. Kumar, "Combinational feature selection approach for network intrusion detection system," in *Proceedings of The Third International Conference on Parallel, Distributed and Grid Computing (PDGC'14)*, pp. 82–87, India, Dec. 2014.
- [14] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, I. H. Witten, "The WEKA data mining software: An update," *ACM SIGKDD Explorations Newsletter*, vol. 11, no. 1, pp. 10–18, 2009.
- [15] B. Ingre, A. Yadav, "Performance analysis of NSL-KDD dataset using ann," in *Proceedings of The*

- International Conference on Signal Processing And Communication Engineering Systems (SPACES'15)*, pp. 92–96, Jan. 2015.
- [16] M. Kumar, K. Dutta, I. Chopra, “Impact of wormhole attack on data aggregation in hierarchical WSN,” *International Journal of Electronics and Information Engineering*, vol. 1, no. 2, pp. 70–77, 2014.
- [17] A. J. Malik, W. Shahzad, F. A. Khan, “Network intrusion detection using hybrid binary pso and random forests algorithm,” *Security and Communication Networks*, vol. 8, no. 16, pp. 2646–2660, 2015.
- [18] J. McHugh, “Testing intrusion detection systems: A critique of the 1998 and 1999 darpa intrusion detection system evaluations as performed by lincoln laboratory,” *ACM Transactions on Information System Security*, vol. 3, no. 4, pp. 262–294, 2000.
- [19] MIT Lincoln Labs, “Darpa intrusion detection evaluation,” June 2015. (<https://www.ll.mit.edu/ideval/data/>)
- [20] NSL-KDD, “NSL-KDD data set for network-based intrusion detection systems,” June 2015. (<https://web.archive.org/web/20150205070216/http://nsl.cs.unb.ca/NSL-KDD/>)
- [21] E. U. Opara, O. A. Soluade, “Straddling the next cyber frontier: The empirical analysis on network security, exploits, and vulnerabilities,” *International Journal of Electronics and Information Engineering*, vol. 3, no. 1, pp. 10–18, 2015.
- [22] M. S. Pervez, D. M. Farid, “Feature selection and intrusion classification in NSL-KDD cup 99 dataset employing svms,” in *Proceedings of 8th International Conference on Software, Knowledge, Information Management and Applications (SKIMA'14)*, pp. 1–6, Dhaka, Bangladesh, Dec. 2014.
- [23] J. R. Quinlan, “Induction of decision trees,” *Machine Learning*, vol. 1, no. 1, pp. 81–106, 1986.
- [24] G. P. Rout, S. N. Mohanty, “A hybrid approach for network intrusion detection,” in *Proceedings of The Fifth International Conference on Communication Systems and Network Technologies (CSNT'15)*, pp. 614–617, Gwalior, India, Apr. 2015.
- [25] Z. Salek, F. M. Madani, R. Azmi, “Intrusion detection using neural networks trained by differential evaluation algorithm,” in *Proceedings of The 10th International Conference on Information Security and Cryptology (ISCISC'13)*, pp. 1–6, Yazd, Iran, Aug. 2013.
- [26] S. Samira, M. Zaiton, A. Idawaty, B. Mehdi, “GA and SVM algorithms for selection of hybrid feature in intrusion detection systems,” *International Review on Computers and Software*, vol. 10, no. 3, pp. 2, 2015.
- [27] M. Sreenath, J. Udhayan, “Intrusion detection system using bagging ensemble selection,” in *Proceedings of The International Conference on Engineering and Technology (ICETECH'15)*, pp. 1–4, India, Mar. 2015.
- [28] G. Stein, B. Chen, A. S. Wu, K. A. Hua, “Decision tree classifier for network intrusion detection with ga-based feature selection,” in *Proceedings of the 43rd Annual Southeast Regional Conference*, vol. 2, pp. 136–141, New York, USA, 2005.
- [29] S. J. Stolfo, W. Fan, W. Lee, A. Prodromidis, P. K. Chan, “Cost-based modeling for fraud and intrusion detection: Results from the jam project,” in *Proceedings of The Conference on DARPA Information Survivability Conference and Exposition (DISCEX'00)*, vol. 2, pp. 130–144, California, USA, Jan. 2000.
- [30] R. Storn, K. Price, “Differential evolution – A simple and efficient heuristic for global optimization over continuous spaces,” *Journal of Global Optimization*, vol. 11, no. 4, pp. 341–359, 1997.
- [31] M. F. Tasgetiren, Q. K. Pan, P. . Suganthan, Y. C. Liang, “A discrete differential evolution algorithm for the no-wait flowshop scheduling problem with total flowtime criterion,” in *Proceedings of The Symposium on Computational Intelligence in Scheduling*, pp. 251–258, Apr. 2007.
- [32] M. Tavallaee, E. Bagheri, W. Lu, A. A. Ghorbani, “A detailed analysis of the KDD cup 99 data set,” in *Proceedings of the Second IEEE International Conference on Computational Intelligence for Security and Defense Applications (CISDA'09)*, pp. 53–58, Piscataway, NJ, USA, 2009.
- [33] X. Z. Wang, “ACO and SVM selection feature weighting of network intrusion detection method,” *Analysis*, vol. 9, no. 4, pp. 129–270, 2015.

Biography

Ebenezer O. Popoola received B.Sc. in Electronic and Electrical Engineering from Obafemi Awolowo University, Ile-Ife, Nigeria. He worked in the telecommunication industry before proceeding for his M.Sc. Degree in Computer Science at the University of KwaZulu-Natal, South Africa. His primary interests are communication security, computer vision, machine learning and optimization.

Aderemi O. Adewumi received the B.Sc. and M.Sc. degrees in Computer Science from the University of Lagos, Nigeria, and PhD in Computational & Applied Mathematics from the University of Witwatersrand, South Africa, with a specialty in optimization and computational intelligence. He is currently with the University of KwaZulu-Natal, Durban, South Africa, where he leads the Optimization and Modeling Research Group in the School of Mathematics, Statistics and Computer Science. His current research interests include optimization and artificial intelligence, with a particular interest in computational intelligence, machine learning, and intelligent solutions to real-world global optimization problems.

GPS Spoofing Detection Based on Decision Fusion with a K -out-of- N Rule

Minhong Sun^{1,2}, Yuan Qin², Jianrong Bao^{1,2} and Xutao Yu¹

(Corresponding author: Minhong Sun)

School of Information Science and Engineering, Southeast University¹

No.2, Sipailou, Nanjing, Jiangsu Province 210096, P. R. China

School of Communication Engineering, Hangzhou Dianzi University²

No. 1, Ave. 2, Xiasha Tertiary Education Zone, Hangzhou, Zhejiang Province 310018, P. R. China

(Email: cougar@hdu.edu.cn)

(Received May 12, 2016; revised and accepted Aug. 3 & Sept. 3, 2016)

Abstract

In order to obtain higher detection probability of the GPS spoofing, a general identification scheme with decision fusion is proposed. Firstly, the singular values of the wavelet transformation coefficients of both spoofing and genuine signal are computed and formed as the feature vectors. Then, the feature vectors are input into three classifiers, which are the support vector machines (SVM), the probabilistic neural networks (PNN) and the decision tree (DT), respectively, for GPS spoofing identification. Finally, the results of the three classifiers are fused with a K -out-of- N decision rule, and the final classification result is obtained. Simulation results exhibit the effectiveness of the proposed scheme, whose detection probability has increased by 3.75%, 5.06% and 12.36% than that of the SVM, the PNN and the DT on average, respectively. Moreover, the false alarm probability of the proposed scheme is lower than that of the three classifiers. In addition, the area under curve (AUC) is given to verify the effectiveness and feasibility of the proposed method.

Keywords: Decision Fusion; Feature Extraction; GPS; Spoofing Detection

1 Introduction

A GPS spoofing is an intentional jamming, which is very similar to a true navigation signal. Spoofing interference deceives a GPS receiver to capture the jamming signal, which may cause serious security problems, such as erroneous synchronization time and false position, or no information output [10]. The key task for GPS receiver against spoofing is to identify it correctly. Existing works focused on the detection of spoofing with many different signal features, such as signal power, pseudorange measurements [13], time of advent, signal parameters estimation [6] etc. However, most of them applied only one clas-

sifier/detector. The utilization of multi-classifiers with a decision fusion technique to further improve the detection performance is ignored.

In this paper, several classifiers have been proposed to detect spoofing attack, including the support vector machines (SVM) [9], the probabilistic neural networks (PNN) [11], and the decision tree (DT) [7], etc. Although each classifier functions well, the detection performance can be further improved. Seeking higher detection probability of spoofing, we focus on the multi-classifiers fusion technique with a K -out-of- N decision rule. Due to high reliability, the K -out-of- N rule has a wide and success utilization in many fields [1]. In this paper, we present a GPS spoofing identification method based on the multi-classifiers fusion. The approach is divided into three steps. Firstly, we extract the singular values of the wavelet transformation (WT) coefficients of a signal as a feature vector. By the singular value decomposition (SVD), the quantity of the calculation can be reduced notably [5]. Secondly, based on the same feature vector, three different classifiers, i.e. SVM, PNN and DT, are adopted in the identification of the spoofing, respectively. Thirdly, with the recognition results of each classifier, final identification result is obtained by decision fusion with the K -out-of- N rule.

The rest of the paper is organized as follows. Section 2 describes the feature extraction steps of the received signals. Section 3 introduces three methods of the classifiers briefly. Section 4 illustrates the decision fusion scheme based on the K -out-of- N rule. Simulations and performance analyses are presented in Section 5. Finally, a brief conclusion is given in Section 6.

2 Features Extraction

Two maps are defined in this section for the process of features extraction. The first uses the WT to map a one-

dimensional (1-D) received signal to a two-dimensional (2-D) time-frequency matrix, and the second maps the 2-D signal to a 1-D vector with SVD.

2.1 WT

Assuming the received signal in a navigation receiver is $x(t)$, we can define a map as

$$f[x(t)] \rightarrow \mathbf{W}$$

where $f[\cdot]$ is a WT operator, \mathbf{W} is a time-frequency matrix and it can be represented as $\mathbf{W} = [\mathbf{d}_1, \mathbf{d}_2, \dots, \mathbf{d}_j, \mathbf{a}_j]$. The columns of \mathbf{W} are represented by

$$\mathbf{d}_j = \mathbf{V}_j x(t) = \frac{1}{\sqrt{2}} \sum_{n \in \mathbb{Z}} g(n) \mathbf{S}_{j-1} x(t - 2^{j-1}n)$$

$$\mathbf{a}_j = \mathbf{S}_j x(t) = \frac{1}{\sqrt{2}} \sum_{n \in \mathbb{Z}} h(n) \mathbf{S}_{j-1} x(t - 2^{j-1}n)$$

where n and j denote the filter and decomposition level, respectively; $h(n)$ and $g(n)$ indicate the low-pass and high-pass decomposition filters, respectively; \mathbf{S}_j and \mathbf{V}_j denote the approximation coefficients and the detail coefficients of level j , respectively.

2.2 SVD

SVD is an effective method to reduce data dimension. Utilizing the SVD of a matrix in computations has the advantage of being more robust to numerical errors. The SVD exposes the geometric structure of a matrix, which is an important aspect in many matrix calculations. Then the second map is defined by the following expression as

$$g[\mathbf{W}] \rightarrow \theta.$$

3 Classifiers

Because of their good classification performance and wide applications, three main approaches, i.e. SVM, PNN and DT, are chosen for the identification of the GPS spoofing in our scheme.

3.1 SVM

The SVM is based on the Vapnik-Chervonenks dimension of statistical learning theory and structural risk minimization inductive principle, which can deal with small samples, non-linear and high dimension pattern recognition problems [12]. Moreover, it does not suffer from overfitting and it has good ability of generalization. The decision function of it can be represented as

$$f(x) = \text{sgn} \left(\sum_{i=1}^l \alpha_i y_i K(\mathbf{x}_i, \mathbf{x}) + b \right)$$

where \mathbf{x}_i is the support vector, $y_i \in \{-1, 1\}$ is the class label, $K(\mathbf{x}_i, \mathbf{x})$ is the kernel function, α_i is the Lagrangian

multiplier, b is the classification threshold, $\text{sgn}(\cdot)$ is the signum function. Comparing with other kernel functions, radial basis kernel function (RBF) has the advantage of higher precision, less parameters, and better performance [12]. Hence, a RBF is applied in our case and it is expressed as

$$K(\mathbf{x}_i, \mathbf{x}) = \exp\{-\|\mathbf{x}_i - \mathbf{x}\|^2 / \sigma^2\}$$

where σ^2 is the kernel parameter.

3.2 PNN

The PNN is a parallel algorithm with supervised learning which uses Bayes decision rule and Parzen window [8]. Especially in the application of solving the classification problems, the superiority of it is obvious. It can use linear learning algorithm to accomplish the work by nonlinear learning algorithm, while it keeps the nonlinear features such as high accuracy of the algorithm.

The output of the output layer can be expressed as

$$\text{if } n_j = \max_k(n_k) \quad y_j = 1 \quad \text{else } y_j = 0$$

where n denotes the output of summation layer, k is the number of samples in training set, j indicates the number of max layers. The probability value n_j corresponding to the maximum is 1, i.e. $y_j = 1$ and the rest values are 0, i.e. $y_j = 0$.

3.3 DT

A decision tree has a flowchart-like tree structure, where each non-leaf node denotes a test on a pattern attribute, each branch represents an outcome of the test, and each leaf node is labeled by a class [3]. Up to now, many approaches have been proposed for decision trees, such as ID3 and C4.5 [3]. We use C4.5 which is an extension of ID3. C4.5 algorithm selects properties by information gain ratio, which is written as

$$\text{GainRatio}(\mathbf{S}, \mathbf{A}) = \frac{\text{Gain}(\mathbf{S}, \mathbf{A})}{\text{SplitInformation}(\mathbf{S}, \mathbf{A})}$$

where \mathbf{S} and \mathbf{A} denote the sample set and the properties, respectively. $\text{Gain}(\mathbf{S}, \mathbf{A})$ and $\text{SplitInformation}(\mathbf{S}, \mathbf{A})$ are the information gain and split information, respectively.

4 Decision Fusion

In order to make a more accurate decision for the spoofing detection, a decision fusion approach is presented and capable of overcoming the disadvantage of single classifier, and eliminating the system uncertainty. The K -out-of- N rule is also selected for the decision fusion.

Each classifier is independent. The binary decision at the i th classifier to decide the real signal or the deceptive

jamming is given by

$$\begin{cases} H_0 : u_i = 0 \\ H_1 : u_i = 1 \end{cases}$$

where $i = 1, 2, \dots, N$, H_0 represents the real signal, H_1 represents the spoofing signal, N is the number of classifiers, and u_i is the output of the i th classifier.

The results above are input into the fusion center with the K -out-of- N rule. The following expression describes the K -out-of- N rule, i.e.

$$\begin{cases} H_0 : u_i = 0 \text{ if } \sum_{i=1}^N u_i < K \\ H_1 : u_i = 1 \text{ if } \sum_{i=1}^N u_i \geq K \end{cases} \quad (1)$$

where u_0 is the final decision. The Equation (1) demonstrates that if the sum of the outputs of N classifiers is larger than or equal to K , the spoofing signal is detected, i.e., H_1 . Otherwise, the received signal is a real one, i.e., H_0 . Then the OR rule corresponds to the case of $K = 1$ and the AND rule corresponds to the case of $K = N$.

The overall performance of detection is evaluated by two indicators, such as the overall detection probability (PD) and the overall false alarm probability (PF). PD and PF are expressed respectively as follows [1]

$$P_D = \sum_{j=k}^N \sum_{\sum u_i=j} \prod_{i=1}^N (P_{di})^{u_i} (1 - P_{di})^{1-u_i}$$

$$P_F = \sum_{j=k}^N \sum_{\sum u_i=j} \prod_{i=1}^N (P_{fi})^{u_i} (1 - P_{fi})^{1-u_i}$$

where P_{di} and P_{fi} represent the detection probability $p(H_1|H_1)$ and false alarm probability $p(H_1|H_0)$ of the i th classifier, respectively.

5 Simulations and Analyses

In this section, the performance of the detection based on decision fusion method is simulated and analyzed, to verify the effectiveness of the proposed algorithm. Both the detection probability and false alarm probability are used in the analyses with numerical computations.

Suppose that a GPS signal is a C/A code signal with QPSK modulation. SNR is set from 2dB to 14 dB with a step of 1dB. Then, the experiments with K -out-of- N rule and three single classifiers are carried out. In order to generate a spoofing jamming signal, which is very similar to a real satellite navigation transmitter, a Hammerstein model is used [2], which is composed of a static nonlinear subsystem followed by a dynamic linear subsystem. A satellite transmitter or a spoofer is regarded as a static nonlinear subsystem, which is modeled as a memoryless polynomial model. The wireless channel is regarded as a dynamic linear subsystem, which is expressed as a FIR filter. The relationship between the input and output of

the whole system is given as

$$y(n) = \sum_{k=0}^{N-1} h_k \sum_{i=1}^M b_{2i-1} |d(n-k)|^{2i-2} d(n-k) + w(n)$$

where M is the number of the polynomial coefficients, $d(n)$ denotes the input signal, b_{2i-1} is the polynomial coefficients, h_k is the channel response coefficient, N denotes the order of FIR filter, $W(n) \sim \mathcal{N}(0, \sigma^2)$ indicates additive Gaussian white noise (AWGN). Different systems are simulated with different vectors of the parameters $[b_{2i-1} h_k]$. Two training sets consisting of 1500 sample signals per class, and two test sets consisting of 500 sample signals per class are generated by the Hammerstein model. Each sample signal contains 500 points.

Two sets of parameters are set and shown in Table 1. One is from the satellite transmitter, and the other is from the spoofer. The two models' parameters are set to be very close to each other, for the spoofing signals are very similar to the real ones.

With the foregoing features extraction method, the feature vectors are calculated. The average singular values of the real signal and the jamming signal are shown in Table 2 for the case of the SNR being 10dB.

For each SNR, 200 independent experiments were run. The outputs of each classifier and the final detection results on the basis of the K -out-of- N rule are obtained. The detection curves are illustrated in Figure 1 and Figure 2.

From Figure 1, we can see that the detection probability is increased with the increasing of the SNR values. The average detection probability of the decision fusion method has increased by 3.75%, 5.06% and 12.36% than that of the SVM, the PNN and the DT, respectively. Hence, cooperation among classifiers can be used in order to improve the reliability of the detection results. From Figure 2, it is evident that the false alarm probabilities of the four methods are lower than 0.1. The average false alarm probability of decision fusion method has decreased by 1.25% than that of the SVM, by 3.70% than that of the PNN, and by 7.28% than that of the DT, respectively. Therefore, the detection performance of the three classifiers is improved by the decision fusion method.

Receiver operating characteristic (ROC) curve is commonly used to characterize the detection performance. However, with this metric, the performance comparison with multiple classifiers would be difficult. An alternative metric, based on the area under the ROC curve (AUC) seems appropriate in this situation. The AUC can be calculated by

$$AUC = \frac{S_0 - n_0(n_0 + 1)/2}{n_0 n_1}$$

where n_0 and n_1 are the numbers of positive and negative samples, respectively, and $S_0 = \sum r_i$, where r_i is the rank of the i th positive example in the ranked list [4].

The larger value of AUC is, the better performance of the classifier will have. The AUC values for the four

Table 1: Parameters configuration

	parameter \mathbf{b} of the nonlinear subsystem				parameter \mathbf{h} of the linear subsystem		
	b_1	b_3	b_5	b_7	h_1	h_2	h_3
Transmitter	1	-0.0735	-0.0986	-0.0547	0.9906	0.0628	0.0079
Spoofers	1	-0.0728	-0.0976	-0.0542	0.9807	0.0622	0.0078

Table 2: Singular values of signals (SNR=10dB)

	Singular Values								
	Real signal	57.43	45.01	36.43	32.50	27.24	23.90	20.95	16.19
Spoofing signal	56.00	43.89	35.51	31.71	26.56	23.30	20.43	15.78	9.24

methods with different SNR are shown in Table 3. By comparing the AUC, we draw a conclusion that the performance of the decision fusion method is better than the other three methods with a single classifier.

Table 3: AUC comparison

AUC	5dB	10dB	15dB
SVM	0.8931	0.9535	0.9801
PNN	0.8357	0.9401	0.9890
DT	0.7962	0.8731	0.9070
K/N	0.9030	0.9672	0.9900

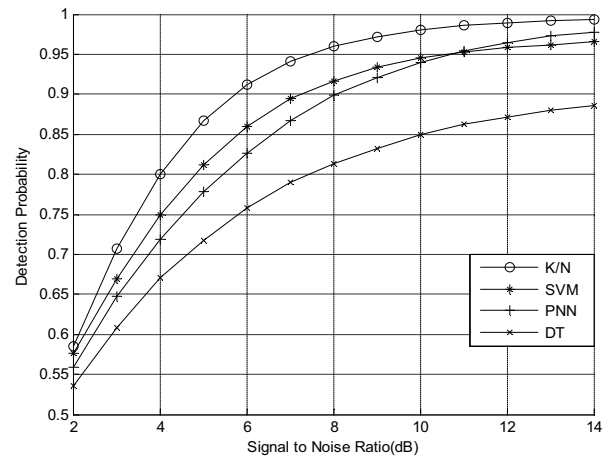


Figure 1: Detection probability comparison between fusion and single classifier

6 Conclusion

We have shown that the overall detection precision of GPS spoofing jamming is improved by using a decision fusion method with the K -out-of- N rule. A spoofing signal is detected if at least K out of N classifiers have made the same decision. As cooperation among classifiers, the reliability of the detection results is improved. Simulation results are presented to demonstrate the effectiveness of the approach, whose detection probability is higher than that of the three classifiers, i.e. SVM, PNN and DT, and the false alarm probability is lower than that of the three classifiers, in the case of SNR ranging from 2dB to 14dB. Furthermore, it is illustrated with AUC that the proposed method is more effective than the methods with only a single classifier.

Acknowledgments

This work was supported by the Natural Science Foundation of China (No.61271214, No.61471152), by the Postdoctoral Science Foundation of Jiangsu Province (No.1402023C) and by the Zhejiang Provincial Natural Science Foundation of China (No. LZ14F010003).

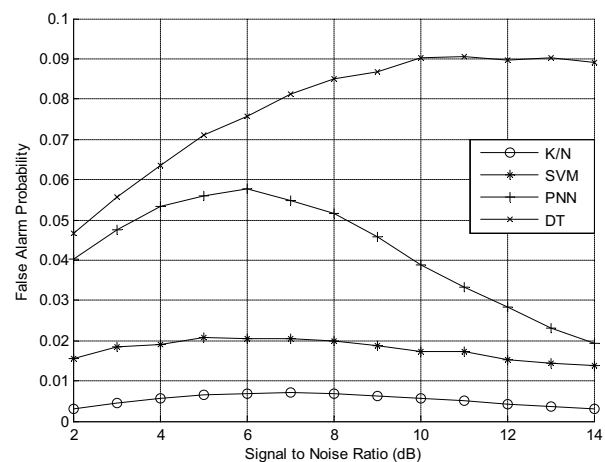


Figure 2: False alarm probability comparison between fusion and single classifier

References

- [1] S. Althunibat, M. D. Renzo, and F. Granelli, "Optimizing the k-out-of-n rule for cooperative spectrum sensing in cognitive radio networks," in *IEEE Global Communications Conference (GLOBECOM'13)*, pp. 1607–1611, Atlanta, USA, Dec. 2013.
- [2] F. M. Barradas, T. R. Cunha, P. M. Lavrador, and J. C. Pedro, "Polynomials and luts in pa behavioral modeling: A fair theoretical comparison," *IEEE Transactions on Microwave Theory and Techniques*, vol. 62, no. 12, pp. 3274–3285, 2014.
- [3] H. W. Chiu, C. S. Ouyang, and S. J. Lee, "Improved c-fuzzy decision trees," in *IEEE International Conference on Fuzzy Systems (Fuzz-IEEE'06)*, pp. 1763–1768, Vancouver, Canada, July 2006.
- [4] D. J. Hand and R. J. Till, "A simple generalisation of the area under the roc curve for multiple class classification problems," *Machine Learning*, vol. 45, no. 2, pp. 171–186, 2001.
- [5] N. C. Kim and H. J. So, "Comments on svd-based modeling for image texture classification using wavelet transform," *IEEE Transactions on Image Processing*, vol. 22, no. 12, pp. 5408–5408, 2013.
- [6] J. Kou, S. Xiong, S. Wan, and H. Liu, "The incremental probabilistic neural network," in *International Conference on Natural Computation (ICNC'10)*, pp. 1330–1333, Yantai, China, Aug. 2010.
- [7] X. Liu and W. Jin, "Performance analysis of bootstrap based distributed detector under correlated k-distributed clutter," in *Asian-Pacific Conference on Synthetic Aperture Radar (AP SAR'09)*, pp. 556–559, Xi'an, China, Oct. 2009.
- [8] S. Mishra, C. N. Bhende, and B. K. Panigrahi, "Detection and classification of power quality disturbances using s-transform and probabilistic neural network," *IEEE Transactions on Power Delivery*, vol. 23, no. 1, pp. 280–287, 2008.
- [9] H. J. Patel, M. A. Temple, and R. O. Baldwin, "Improving zigbee device network authentication using ensemble decision tree classifiers with radio frequency distinct native attribute fingerprinting," *IEEE Transactions on Reliability*, vol. 64, no. 1, pp. 225–230, 2015.
- [10] M. L. Psiaki, "Gps spoofing detection via dual-receiver correlation of military signals," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 49, no. 4, pp. 2250–2267, 2013.
- [11] Y. Yan, Q. Tian, and Y. Wang, "Performance analysis of detection algorithm for follower noise jamming in nakagami fading channels," *Chinese Journal of Radio Science*, vol. 29, no. 4, pp. 738–744, 2014.
- [12] H. Yang, X. Xie, and R. Wang, "Som-ga-svm detection based spectrum sensing in cognitive radio," in *Wireless Communications, Networking and Mobile Computing (WiCOM 2012)*, pp. 1–7, Shanghai, China, Sept. 2012.
- [13] J. Yang, Y. Chen, W. Trappe, and J. Cheng, "Detection and localization of multiple spoofing attackers in wireless networks," *IEEE Transactions on Parallel and Distributed Systems*, vol. 24, no. 1, pp. 44–58, 2013.

Biography

Minhong Sun received his Ph.D. E.E. degree from the Department of Electronic Engineering, University of Electronic Science and Technology of China (UESTC), Chengdu, China, in 2008. He is with the school of Communication Engineering, Hangzhou Dianzi University, Hangzhou, China. His main research interests include signal processing and information countermeasures.

Yuan Qin received her M.S.E.E. degree from Hangzhou Dianzi University, Hangzhou, China, in 2016. Her main research interests include machine learning and information countermeasures.

Jianrong Bao received his B.S. degree in Polymeric Materials & Eng., and the M.S.E.E. degree both from Zhejiang University of Technology, Hangzhou, China, in 2000 and 2004, respectively. He received his Ph.D. E.E. degree from the Department of Electronic Engineering, Tsinghua University, Beijing, China, in 2009. He is with the school of Information Engineering, Hangzhou Dianzi University, Hangzhou, China. His main research interests include wireless communication and so on.

Xutao Yu received the BS and MS degrees from Hohai University, Nanjing, China in 1997 and 2000, respectively, and the PhD degree from Southeast University, Nanjing, China in 2004, all in communication and information systems. Since 2004, she has been with the the State Key Laboratory of Millimeter Waves and is currently a professor there. She has published over 80 papers and issued over 20 patents. Her research area is wireless communication.

Design and Implementation of an Intrusion Prevention System

Yousef Farhaoui

(Corresponding author: Yousef Farhaoui)

ASIA Team, Department of Computer Science, Faculty of Sciences and Technique, Moulay Ismail University
B.P 509, Boutalamine, Errachidia, Morocco
(Email: youseffarhaoui@gmail.com)

(Received June. 03, 2016; revised and accepted Aug. 21 & Sept. 5, 2016)

Abstract

In view of the recent advances of communication and information technology along with the growing need for on-line networking, computer security has become a challenge to almost all the studies that have been carried out in this research axis. So far, various tools and mechanisms have been developed in order to guarantee a safety level up to the requirements of modern life. Among these, intrusion detection and prevention systems (IDPS) tend to locate activities or abnormal behaviors suspect to be detrimental to the correct operation of the system. In this respect, this work targets the design and the realization of an IDPS inspired from natural immune systems. The immune systems have aroused the interest of researchers in the intrusion detection field, taking into account the similarities of NIS (Natural Immune System) and IDPS objectives. Within the Framework of this work, we conceived an IDPS inspired from natural immune system and implemented by using a directed approach. A platform was developed and tests were carried out in order to assess our system performances.

Keywords: Artificial Immune System; Intrusion Detection System; Intrusion Prevention System; Security Systems

1 Introduction

Since their appearance, computer attacks have been a real threat. With their great diversity and specificity to systems, these can have catastrophic consequences. Various measures to prevent these attacks or reduce their severity exist but there is no complete solution.

The IPS is one of these current most effective measures. Their role is to recognize intrusions or intrusion attempts by abnormal users' behaviors, or recognize attacks from the network data stream. Different methods and approaches have been adopted for the design of IPS, most significantly, the methods inspired by nature, espe-

cially the immune system [12, 13, 15], which has properties and great similarity to IPS. The study of the immune system is a promising new area of research (artificial intelligence), namely, artificial immune systems (AIS) [4, 28]. These are actually modelling implementation and adaptation of concepts and methods of the biological immune systems to solve problems. Our goal is to develop an artificial immune system for our intrusion prevention system, implementing the main immune theories. To evaluate performance, we will conduct a series of tests to analyze the results in order to measure the contribution of the immune systems in the intrusion prevention [9, 22].

2 Natural Immune Systems (NIS)

2.1 NIS Properties

The NIS is a source of inspiration for new branches of IT. With very important properties, it has become a valuable reference. Several research works have been developed on this basis. The most important property which is the basis of immune reactions is the ability of the NIS to distinguish between self cells and non-self cells and the ability to recognize the exact type of each foreign cell [2, 9, 22]. In each contact with a new kind of antigens, the NIS categorizes it and keeps it in mind, thanks to a cell division mechanism followed by a selection process to refine and improve the response of NIS in the next contact with the same antigen. This allows the NIS to increase efficiency to the recognition of antigens; this process is called affinity maturation [3]. The different actors of NIS need to exchange messages under the form of signals. This occurs by means of two types of dialogues: the one-way dialogues by the immunological components and the continuous dialogues through an exchange of molecular signals [26].

2.2 Immune Theories

The behavior and reactions of the NIS are primarily governed by immune theories. This theory manages the pro-

cess of creating cells. In particular, it manages the creative process at the level of the discrimination between self and non-self cells. Lymphocytes have receptors on their surfaces. Lymphocytes from the bone marrow migrate to the thymus, at this stage they are called immature or naïve T cells. Their para-topes undergo a process of pseudo-random genetic rearrangement. After that, a very important test is introduced [1, 7]. The recognition of an antigen by B cells, which produce specific antibodies. The antibody associated with the antigen using receptor then using cells such as T aide uses, B cells of stimulated and a proliferation process allows B cells to reproduce by creating clones themselves [6]. A second process will select among those new cells with a high affinity to make memory cells [19].

3 Artificial Immune Systems

Artificial Immune Systems (AIS) is a new branch of artificial intelligence. Inspired from remarkable properties and concepts of biological immune system [4], AIS are designed to solve various problems. They are a mathematical or computer implementation of the operation of the natural immune system.

3.1 Modelling AIS

The common model known as the Framework of AIS defines the rules to be complied by AIS and the process to develop new approaches. The necessary conditions are [5]: The representation of the system components. Adapting procedures to monitor the evolution of the system. The two conditions mentioned above are imperative for the development of a framework to define AIS [3]. Then, the form of an antibody consists of a set of l parameters. These parameters may be represented by a point in a space of l dimensions. A first notes that in this plan, those antibodies are close to each other. Population or repertoire of N individuals is modelled as a space forms a finite volume V containing N points. An antigen is represented by the point $Ag = \langle Ag_1, Ag_2, \dots, Ag_l \rangle$, an antibody is also represented by a point $Ab = \langle Ab_1, Ab_2, \dots, Ab_l \rangle$. To measure the degree of completeness between the antigen and the antibodies, several techniques can be used. Most often the distances are used [17]:

Euclidean distance

$$D = \sqrt{\sum_{i=1}^l (Ab_i - Ag_i)^2}$$

Manhattan distance

$$D = \sum_{i=1}^l \|Ab_i - Ag_i\|$$

Hamming distance

$$D = \sum_{i=1}^l \delta_i \text{ with } \left\{ \begin{array}{ll} 1 & \text{if } Ab_i \neq Ag_i \\ \delta_i = 0 & \text{if not} \end{array} \right\}$$

if $D \uparrow \Rightarrow$ Affinity \downarrow .

So, we notice that the antigen-antibody affinity is relative to the distance in the space between them. Once the antigens and antibodies are represented, the quantitative function of the defined Completeness degree between them, it remains only to implement the immune theories.

3.2 Immune Algorithms

The Algorithm 1 Show how immune theory work. This theory is based on the principle that only the cells having the antigen recognize the antigen proliferate and become memory cells. The clonal selection algorithm is based on the following processes:

- Holding a set of memory cells;
- Selection and cloning of the most stimulated antibodies;
- Re-selection clones proportionally to the affinity with the antigen;
- Removal of unstimulated antibodies.
- The maturation of their affinity [3].

Algorithm 1 Clonal selection algorithm

- 1: Begin
 - 2: P = set of shapes to be recognized
 - 3: M = Population random individuals
 - 4: **while** A minimal form is not recognized **do**
 - 5: **for** $i = 1$ to size of(P) **do**
 - 6: $aff = affinite(P_i, M_i)$.
 - 7: **end for**
 - 8: Select n_1 elements having the best affinity with the elements of M .
 - 9: Generate copies of these elements in proportion to their affinity with the antigen.
 - 10: Mutate all copies proportionately with their.
 - 11: Add mutated individuals in the population M .
 - 12: Choose n_2 of these mutated elements(optimized) as memory.
 - 13: **end while**
 - 14: End
-

This concept is very interesting, especially for systems monitoring applications and detection and prevention of abnormal or unusual uses [5]. The problem of protection of computer systems in the learning problem of distinguishing between self and non-self. Rather, they compare the loads detection problem within the systems to the process of adverse selection which takes place in the thymus [25].

The algorithm 2 illustrates a summary of the negative selection algorithm.

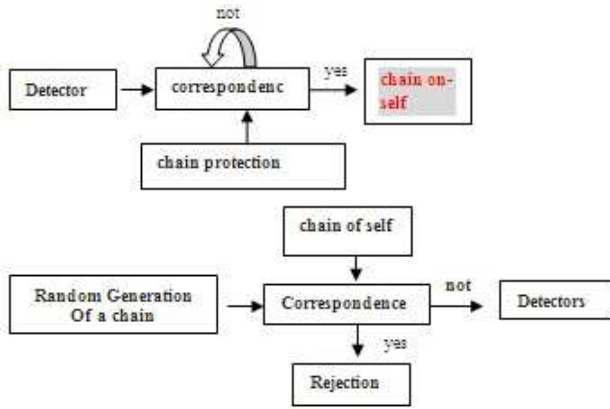


Figure 1: The method of negative selection

Algorithm 2 Negative selection algorithm

- 1: Begin
- 2: S = set of elements of the self.
- 3: D = a detector array.
- 4: SeuilAff = affinity threshold.
- 5: **while** i < nb Detectors **do**
- 6: Generating a d_i detector so that it has no affinity with a member S.
- 7: **if** affinity(d_i , S_i) > SeuilAff **then**
- 8: classified S_i as non-self.
- 9: **else**
- 10: classified S_i as self.
- 11: **end if**
- 12: **end while**
- 13: return a set of detectors D
- 14: End

3.3 Immune Systems Intrusion Detection and Prevention Systems (IDPS)

It is important to recall the functions or the fundamental properties that must satisfy an IDPS as listed in [13, 15]. We will, eventually, try to see what is offered in the parallel artificial immune systems and make the analogy between all IDPS [10, 12, 14]:

Robust: The IDPS must have different points of detection and prevention, and should be highly resistant to attacks.

Configurable: The IDPS must be easily configurable based on each machine on which it will be deployed. The degree of dependence on the operating system must be minimized.

Expandable: Adding new hosts in all machines must be elementary monitored and the dependence on operating systems should not be an obstacle to this extension.

Upgradable: It is necessary that the IDPS can face an unexpected increase in the flow of data to be mon-

itored due to an extension of all the constituents' hosts the IDPS.

Adaptable: The IDPS must dynamically adapt to changes (hardware or software) within the network in question.

Effective: The IDPS should be simple and easy to be deployed in order to avoid affecting the hosts and network performance monitoring.

Distributed: Special attacks can be detected and stopped after the analysis of different signals and alarms from different hosts [24]. The IDPS should be able to recover various events from different stations on the network, analyze them and send responses to different stations. In order to develop an effective IDPS, we will try to find the properties mentioned above in an artificial immune system.

Table 1: Comparing immune systems and immune algorithms

Immune Systems	Immune algorithms
Antigen	Problem to besolved
Antibody	Vector better solutions
Recognition of antigens	Identifying the Problem
Production of antibodies from memory cells	Loading previously best solutions found
Removal of T cells	Elimination of surplus solutions potential
Proliferation of antibodies	Use of aprocessfor creating exact copies of the solution

The immune system is capable of protecting the human body surface from bacteria, viruses or any kind of antigens. This fundamental role is mainly based on the discrimination between self and non-self. The three most important properties of an IDPS are found in the immune systems. The immune systems are [16, 30]. This article talks about the negative selection algorithm. As illustrated in the Figure 1 the algorithm proceeds in two phases :the first is to generate a set of sensors and the second is to use these detectors to monitor data by making a comparison. The comparison can be a comparison of the number of common bits [18, 25, 29]. Once we have found the necessary properties for our IDPS and the choice of using immune systems has been done, it is interesting to have a method for creating algorithms composed of AIS. As illustrated in the Table 1 a comparison between the components of the immune systems and their equivalents in immune algorithms allows us to easily design the algorithms forming our artificial immune system components.

By following this process, we can develop the immune algorithm. This comparison applies to the different problems. We will be interested only in the design of an

IDPS inspired immune systems. The Table 2 shows a very adapted comparison.

Table 2: . Comparing immune systems and IDPS

Immune Systems	IDPS
Thymus and bone marrow	Primary IDPS(supervisor)
Lymphnode	Lymphnode Local Host
Antibody	Detector
Antigen	Intrusion
SelfSelf	Normal activity
Noself	Noself Abnormal activity (suspicious)

Based on this comparison, AIS for detection and intrusion prevention are proposed. These AIS consist of a primary IDPS which acts as a supervisor and a plurality of second IDPS will be installed on each host in the network. The functioning of this IDPS model is as follows: These two points are crucial in creating a detector. Once the elements constituting the detector are listed with the type of each of them, the last step will be to define the values of each detector element as follows. If the item is a continuous type, it will be represented by an interval defined by two terminals. Once the elements and their respective values have been listed, the detector will be represented by a data structure containing these elements [11, 20, 21, 23, 27]. The choice of the clonal selection theory for scenario approach has been made because in this process, this theory is used to generate and refine antibody for the detection of known antigens. We could compare The clonal selection theory, antibodies and antigens detectors known to attack signatures. To conclude, this is the most frequent use of immune theories for the design of intrusion detection systems: NIDPS with detection by scenario, theory of clonal selection HIDPS with behavioral detection and theory of negative selection.

4 Solution Description and Global Architecture of the IDPS Results

We opted for the design of a hybrid IDPS composed of an NIDPS based on the approach of analysis by scenario, implementing the theory of clonal selection and using a signature database and a HIDPS based on behavioral approach, implementing the theory of negative selection and using a user profile database. Using immune theories, the core of our IDPS generates some varied signatures of attacks and user profiles in a pseudorandom manner. This methodology allows us to develop the analyzer to possibly discover new attacks or variants of attacks.

Our IDPS is then composed of:

NIDPS: Generating sensors on the basis of signatures. These detectors will be used to analyze the network traffic.

HIDPS: Based on profiles of normal user's behavior in order to generate detectors able to recognize unusual behaviors of users.

Administration console: From this console, the administrator can configure the different parameters of the IDPS, see the different alerts and start learning control. The components of our solution to be deployed are illustrated in the Figure 2 and are described as following: The NIDPS will be installed on the machine that is the network proxy to analyze all network packets. While, HIDPS will be deployed on all machines that constitute the LAN. Here is the overall architecture of our solution.

5 Databases Used

A large amount of information is analyzed and generated by the various components of our IDPS : the user's profiles, the alerts by the various detectors or the list of attack signatures. The use of databases is very important in the architecture of our IDPS. We opted for the use of three databases.

5.1 The Database "profiles"

This database contains all information about user profiles. The data contained in the database are generated by the HIDPS during the learning phase. For security reasons, user profiles must pass through the HIDPS supervisor to ensure compliance and consistency of the data in the profile.

5.2 The Database "signatures"

This data source is very important; it is the basis of NIDPS. It includes all the known attacks using a certain format. The format of the signature is important insofar as all detectors adopt this format. Unfortunately, there is no standard model for the codification of signatures. The signature must represent a reliable, unambiguous and accurate attributes that can recognize the attack. We must remember that the signatures will be used to analyze the network traffic. It is necessary to define the set of attributes to be used from the set of existing attributes [8, 18]. We propose in this paper a particular model of signatures. Our signature model is designed to meet the requirements by an attack signature. The attack signature must unambiguously represent the attack and should only contain information that allows recognizing the attack. In our case, the signatures are coded so as to be modifiable and can model the new attacks, with new analytical methods... etc. The analysis and synthesis of various network attacks has allowed to classify these into three classes:

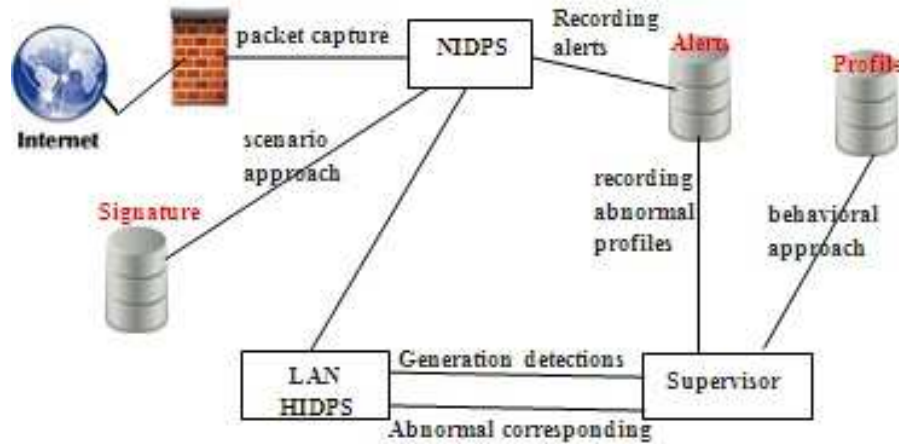


Figure 2: Global solution diagram

Attacks 'data': These are all recognized attacks by analyzing the data portion of the packet, such as SQL Injection attacks. These will be recognized if the following channels (" - , or 1 = 1) is found in the packet.

Attacks 'Headers': These are all recognized attacks by analyzing packet headers, such as DOS attacks with spoofing headers.

Attacks 'Requests/queries': The requests generally include several packages. Some attacks will be recognized by analyzing the set of packets that make up the request, such as attacks of input validation or buffer overflow attacks, which cannot be recognized, that the length or the number of parameters which constitute the request.

In the modelling of different classes of existing attacks, our Signature contains the following fields:

Id	type	Action	Data	Val	Flag
----	------	--------	------	-----	------

- Id: unique identifier of the signature.
- Type: header, data, queries/Request.
- Action: The Action Analysis (e.g., find a sub string, count the number of attributes, length of a query requested service... etc.)
- Data: In the case of attributes kind of strings: the desired string.
- Val: In the case of attributes to numeric values: the value of the attribute.
- Flag: Additional information.

The identifier serves as an index in the signatures database while the type allows to find the table that contains the signature. The action defines the processing to

be used. This is the most important field for a signature. It contains a keyword that shows which method known for analyzing data.

5.3 The Database "Alerts"

This database will list all the alerts generated by the detectors of the two components of IDPS (NIDPS and HIDPS). Each alert should inform the administrator about suspicious event, providing enough information: time, date, sensor, signature or abnormal behavior, the attacker, the victim. This database will be accessed by the administrator to identify traces of attacks or anomalous behavior.

6 HIDPS with Behavioral Approach

The first stage of deployment HIDPS is undoubtedly the learning step, during which it traces back to normal user's behavior by creating a profile for each. User profiles are a source of data that can tell us about the behavior of the users. We chose to use the following information to model a user profile:

- Name of the user;
- Root directory;
- Average consumption CPU and RAM;
- Opening time/closing sessions.

Other information could be used, such as the average consumption of bandwidth, most visited websites, the response speed to the operating system messages.

6.1 Architecture HIDPS

Our HIDPS will consist of a HIDPS supervisor and a plurality of HIDPS slaves to be deployed throughout the network components machinery. The theory of negative selection is the HIDPS core. This theory runs in two phases: the generation of detectors and attack prevention and behavior analysis. The first phase runs on the HIDPS supervisor, which sends alarms generated at HIDPS slaves to execute the second phase of the theory. This consists of analyzing the actual behavior of the user on the basis of sensors.

6.2 HIDPS Supervisor

HIDPS the supervisor's role is to:

- Extract the users of the database profiles.
- Generate detectors and send them to HIDPS slaves by running the first phase of the theory of negative selection those generating sensors that gather all the necessary information for the analysis of user behavior in the future.
- Analyze the HIDPS of reports slaves and list alerts in a database.
- Send commands to start the learning phases, analysis, launch and stopping HIDPS slaves.

6.3 HIDPS Slaves

The main role of HIDPS slaves role is to:

- Generate user profiles during the learning phase.
- Run the second phase of the theory of negative selection, which involves using sensors generated by the first phase in order to analyze the behavior of the user.

6.4 Theory of the Negative Selection

Our HIDPS is based on this theory; it can generate alarms from the user profile and set up at the end to recognize suspicious behaviors. As we have previously seen, this theory runs in two phases:

Phase I: Generation of detectors.

This stage runs on the HIDPS supervisor. During this phase, we extract the user profiles from the database. Each profile will be considered the self system and will be used for the random generation of detectors. Then, a test is set up to purge all alarms generated by keeping only those who do not recognize the self-chain. This phase is shown in Figure 3.

Phase II: Analysis.

This phase runs on HIDPS slaves. During this phase, we operate the detectors generated by the proceeding

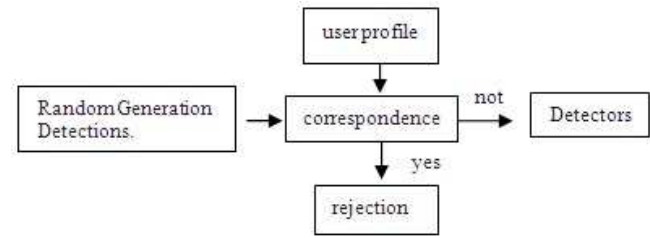


Figure 3: Phase I of negative selection (generation of detections)

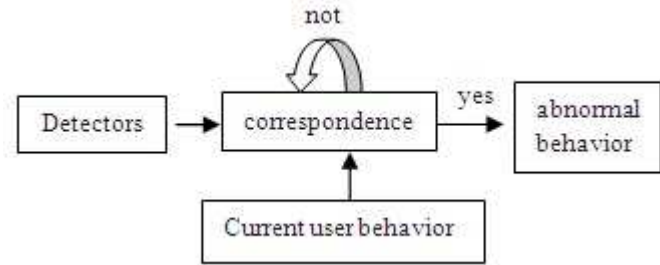


Figure 4: Phase II of negative selection

phase to conduct the analysis of the current behavior of the user. The HIDPS slave must have sensors to inform it about the current behavior of the user. A function will measure the degree of resemblance between that conduct and the detectors previously generated, then an alert is generated if it reaches a certain percentage. This phase is shown in Figure 4.

6.5 Operation HIDPS

As it is clear in the Figure 5, the HIDPS are deploying and starting in two phases:

Learning phase: The HIDPS supervisor sends the command from the beginning of the learning phase for different HIDS slaves. During the learning phase, the HIDPS slave periodically retrieves the user's behavior information.

Monitoring Phase: During this phase, the supervisor HIDPS extracts the profiles of each user, applies the first phase of the negative selection theory to generate detectors. Detectors will be sent to each slave HIDPS with the start command of the monitoring phase.

7 NIDPS with Scenario Approach

The second important component is the NIDPS using analysis with scenario approach. This approach requires a database of known attack signatures on the basis of these signatures, the core of NIDPS generates detectors, can

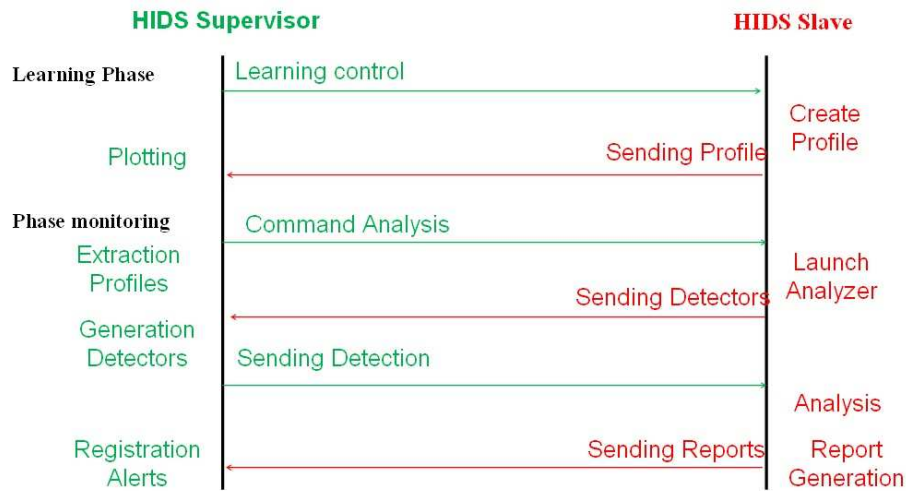


Figure 5: Mode of operation HIDPS

recognize the original signature, but also the signatures derive from the latter. The NIDPS core contains mainly the analysis function; it is based on the theory of clonal selection. The function analysis of our NIDPS contains both detectors generating process and their introduction to the packet-flow analysis.

7.1 Architecture NIDPS

7.1.1 Manager

This is the manager of the solution. The manager is responsible for:

- Starting the different components.
- Assigning different analysis tasks.
- Extracting attack signatures and generating detectors, performing clonal selection algorithm.
- Receive reports and list alerts.

7.1.2 Sensor

The sensor is responsible for capturing the network packets. Different 'sensors' can be deployed in our solution to make this task lighter. If one opts for the deployment of several 'sensors', he must define for each the subset of network traffic that will capture (eg TCP, UDP, ... etc.).

7.1.3 Analyzer

The analyzer is actually comparable to an antibody which is tasked to monitor and recognize certain types of antigens. In our case, the antigen in question is the attack signature to recognize. So the analyzer receives the signatures of the 'Manager' and puts in place to recognize a type of attack. We opt for the joint use of 'Analyzers Sensors'. This use guarantees a lighter and autonomous solution.

7.2 Operation NIDPS

As illustrated in the Figure 6, our analysis uses NIDPS with the scenario approach based on the theory of clonal selection. It is used as a source of data network packets. Here are the steps for its implementation:

Packet capture: The first step of the analysis is capturing the packets through the 'sensors' that capture and transmit the network packets to 'analyzers' to conduct the analysis. At this level, one can also save the captured packets in data structures to analyze them later if the administrator opts for deferred analysis.

Extraction and formatting attributes: This step allows you to extract a high level of attribute vector from the captured packets to be analyzed later. This step is very important. It helps to prepare the packages for the analysis phase by making some changes on them.

Attribute analysis: Once the 'Manager' has generated a set of detectors by applying the theory of clonal selection, the analysis function performed by the Analyzer 'compares to the type of signature, a set of detectors with the attributes of packets. Based on this comparison, many reports are generated.

8 Conclusions

The objective of this work was to design and implement an IDPS inspired for immune systems. The IDPS is a very important brick in any security system. Several research studies using different methods and approaches have been devoted to these. Among these, the artificial immune systems, inspired by the natural immune systems, can be very interesting for the field of intrusion detection, given the similarity of features and objectives of the latter. We

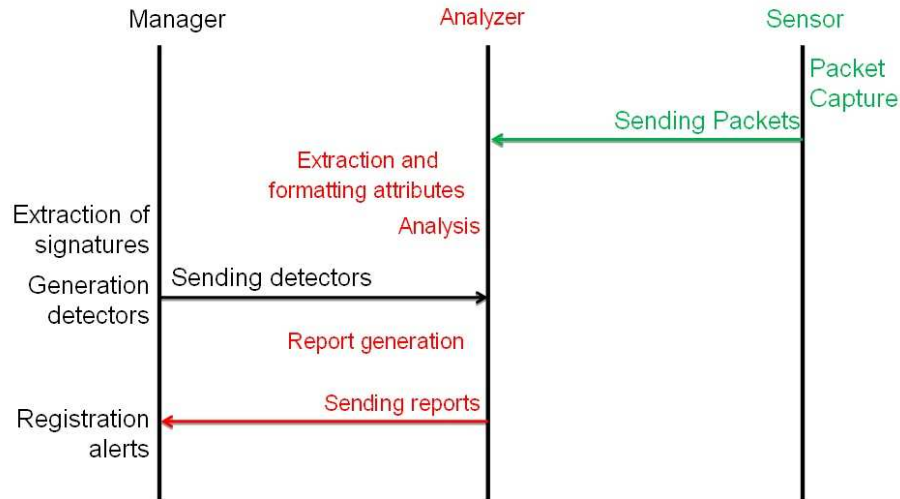


Figure 6: Mode of operation NIDS

focused on the two main theories that are the basis of the immune response, namely the theory of clonal selection and the theory of negative selection. The study of these two immune theories, in the case of intrusion detection, shows that the theory of clonal selection is more appropriate for the scenario analysis, while the theory of negative selection is more appropriate for behavioral analysis. The choice of implementing an IDPS is very important, especially if one considers that the IDPS will be deployed on a network with multiple machines with different hardware and software configurations. As a matter of fact, the IDPS is designed hierarchically and can be distributed across multiple machines, so it requires the analysis of data from different sources. Accordingly, we have designed a hybrid IDPS (NIDPS + HIDPS), analyzing the two sources of information and using both immune theories.

References

- [1] M. Aljabr, "Using classification algorithms in building models for network intrusion detection," *International Journal on Numerical and Analytical Methods in Engineering*, vol. 3, no. 3, pp. 57–62, 2015.
- [2] K. Boukhdar, A. Boualam, S. Tallal, H. Medromi, and S. Benhadou, "Conception, design and implementation of secured uav combining multi-agent systems and ubiquitous lightweight idps (intrusion detection and prevention system)," *International Journal on Engineering Applications*, vol. 3, no. 1, pp. 1–5, 2015.
- [3] J. Brownlee, "Clonal selection theory & clonal selection classification algorithm," *Master of Information Technology, Swinburne, University of Technology*, 2004.
- [4] L. N. De Castro, "An introduction to the artificial immune systems," in *Handbook of Natural Computing*, pp. 1575–1597, 2012.
- [5] L. N. De Castro and J. I. Timmis, "Artificial immune systems as a novel soft computing paradigm," *Soft Computing*, vol. 7, no. 8, pp. 526–544, 2003.
- [6] L. N. De Castro and F. J. Von Zuben, "Learning and optimization using the clonal selection principle," *IEEE Transactions on Evolutionary Computation*, vol. 6, no. 3, pp. 239–251, 2002.
- [7] L. N. De Castro, F. J. Von Zuben, and G. A. de Deus, "The construction of a boolean competitive neural network using ideas from immunology," *Neurocomputing*, vol. 50, pp. 51–85, 2003.
- [8] K. D. D. Cup, *Data/the UCI KDD Archive, Information and Computer Science*, University of California, Irvine, 1999.
- [9] L. N. de Castro and J. Timmis, "Artificial immune systems: A novel paradigm to pattern recognition," *Artificial Neural networks in Pattern Recognition*, vol. 1, pp. 67–84, 2002.
- [10] F. S. de Paula, L. N. de Castro, and P. L. de Geus, "An intrusion detection system using ideas from the immune system," in *IEEE Congress on Evolutionary Computation (CEC'04)*, vol. 1, pp. 1059–1066, 2004.
- [11] M. Enshaei, Z. M. Hanapi, and M. Othman, "A review: Mobile ad hoc networks challenges, attacks, security, vulnerability and routing protocols," *International Journal on Communications Antenna and Propagation*, vol. 4, no. 5, pp. 168–179, 2014.
- [12] Y. Farhaoui and A. Asimi, "Performance assessment of the intrusion detection and prevention systems: According to their features: the method of analysis, reliability, reactivity, facility, adaptability and performance," in *The 6th IEEE International Conference on Sciences of Electronics Technologies Information and Telecommunication (SETIT'12)*, 2006.

- [13] Y. Farhaoui and A. Asimi, "Performance method of assessment of the intrusion detection and prevention systems," *International Journal of Engineering Science and Technology*, vol. 3, no. 7, 2011.
- [14] Y. Farhaoui and A. Asimi, "Model of an effective intrusion detection system on the LAN," *International Journal of Computer Applications*, vol. 41, no. 11, pp. 26–29, 2012.
- [15] Y. Farhaoui and A. Asimi, "Performance assessment of tools of the intrusion detection/prevention systems," *International Journal of Computer Science and Information Security*, vol. 10, no. 1, pp. 7, 2012.
- [16] Y. Farhaoui and A. Asimi, "Performance assessment of tools of the intrusion detection and prevention systems," in *The 3rd IEEE International Conference on Multimedia Computing and Systems (ICMCS'12)*, pp. 1–6, Morocco, Aug. 2012.
- [17] M. Gharbi, "Systèmes immunitaires artificiels et optimisation," *Centre Européen de Réalité Virtuelle*, 2006.
- [18] A. P. Gopi, E. S. Babu, and C. N. Raju and S. A. Kumar, "Designing an adversarial model against reactive and proactive routing protocols in manets: A comparative performance study," *International Journal of Electrical and Computer Engineering*, vol. 5, no. 5, 2015.
- [19] S. A. Hofineyr and S. Forrest, "Immunity by design: An artificial immune system," in *Proceedings of Genetic and Evolutionary Computation Conference*, pp. 1289–1296, 1999.
- [20] L. Jie, W. Ying, and W. F. Chen, "An improved privacy protection security protocol based on NFC," *International Journal of Network Security*, vol. 19, no. 1, pp. 39–46, Jan. 2017.
- [21] G. R. Kavitha and T. S. Indumathi, "Novel roadm modelling with wss and obs to improve routing performance in optical network," *International Journal of Electrical and Computer Engineering*, vol. 6, no. 2, pp. 700, 2016.
- [22] H. Khelil, A. Benyettou, and A. Belaïd, "Application du systme immunitaire artificiel pour la reconnaissance des chiffres," in *Maghrebian Conference on Software Engineering and Artificial Intelligence (MCSEAI'08)*, 2008.
- [23] J. Kim and P. J. Bentley, "An evaluation of negative selection in an artificial immune system for network intrusion detection," in *Proceedings of the 3rd Annual Conference on Genetic and Evolutionary Computation*, pp. 1330–1337, 2001.
- [24] J. Kim, P. J. Bentley, U. Aickelin, J. Greensmith, G. Tedesco, and J. Twycross, "Immune system approaches to intrusion detection—a review," *Natural Computing*, vol. 6, no. 4, pp. 413–466, 2007.
- [25] F. J. Von Zuben L. N. De Castro, "Artificial immune systems: Part i—basic theory and applications," *Universidade Estadual de Campinas, Dezembro de, Tech. Rep.*, vol. 210, no. 1, 1999.
- [26] M. M. Mantha, *The Truth about your Immune System: what you Need to Know*, Harvard College, États-Unis, 2004.
- [27] B. Meng, C. T. Huang, Y. Yang, L. Niu, and D. Wang, "Automatic generation of security protocol implementations written in java from abstract specifications proved in the computational model," *International Journal of Network Security*, vol. 19, no. 1, pp. 138–153, 2017.
- [28] T. M. Mubarak, M. Sajitha, G. A. Rao, and S. A. Sattar, "Secure and energy efficient intrusion detection in 3d wsn," *International Journal on Information Technology*, vol. 2, no. 2, pp. 48–55, 2014.
- [29] Y. Qiao, *An Intrusion Detection System Based on Immune Mechanisms*, SPIE Newsroom, 2007.
- [30] M. Zielinski and L. Venter, "Applying similarities between immune systems and mobile agent systems in intrusion detection," in *ISSA*, pp. 1–12. Citeseer, 2004. (<http://icsa.cs.up.ac.za/issa/2004/Proceedings/Full/016.pdf>)

Biography

Yousef Farhaoui is an professor, Department of Computer Science in Faculty of sciences and Techniques, Moulay Ismail University, Morocco. Received his PhD degree in computer security from the University Ibn Zohr. His research interest includes computer security, Data Mining, Data Warehousing, Data Fusion etc..

Securing Portable Document Format File Using Extended Visual Cryptography to Protect Cloud Data Storage

K. Brindha and N. Jeyanthi

(Corresponding author: K. Brindha)

School of Information Technology and Engineering, VIT University

Vellore 632014, Tamilnadu, India

(Email: brindha.k@vit.ac.in)

(Received Apr. 1, 2016; revised and accepted July 19 & Aug. 29, 2016)

Abstract

With the vast development in cloud computing model, various organizations and individuals often deploy the cloud without reviewing the security policies and procedures which can cause great risk in their business. Securing data in cloud storage becomes a challenging task not only for the cloud user but also to the Cloud Service Provider (CSP). Storing secret data in unencrypted form is susceptible to easy access to both the unauthorized people and the CSP. Standard encryption algorithms require more computational primitives, storage space and cost. Therefore protecting cloud data with minimal computation and storage space is of paramount significance. The Securing Portable Document Format file Using Extended Visual Cryptography (SPDFUEVC) technique proposes efficient storage to achieve data confidentiality and integrity verification with minimal computation, time complexity and storage space.

Keywords: Confidentiality; Cryptography; Data Integrity; Visual Cryptography

1 Introduction

Leading to dramatic change in the computing services, cloud computing has become very popular in IT industries. It attracts the attention of industry and academia alike. The main aim of cloud computing is to provide flexible, feasible and secure services to users of network [10, 21, 27, 37]. In the present revolutionary scenario of IT explosion, cloud computing faces the ever growing demand for large scale computing with minimum cost and fast networking technologies. It has to prove its economic feasibility both in terms of setup and maintenance [24, 39]. A cloud provides fundamental services such as Infrastructure as a Service (IaaS), Platform as a Service (PaaS) and Software as a Service (SaaS) to the customers. Even

though the benefits of using the cloud are clear and understood, some of the problems persist and remain unsolved [36, 45, 46].

Enterprises and individuals who have stored their files on the cloud storage are worried neither about the storage space in the hard disk of the computer nor the risk of the loss of their valuable files due to computer crash. Different CSPs like Google, Amazon, Apple, Microsoft, etc. offer various storage services to customers for storing their valuable files safely on the external storage terms and cost of storage services and other special benefits vary from one service provider to another. They offer certain amount of free storage space to customers. Usually cloud consumers can safely store their files in One drive, Google drive, Sky drive, Drop box, etc. with their personal email address and password. PDF (Portable Document Format) is a file format that has in itself all the elements of a printed document as an electronic image. It is especially useful for documents such as medical records, financial data, tender quotations etc. Hence most of the potential users store their information in the PDF file but in unencrypted form. Security as the major threat to the above cloud storage system [43].

The issue of security in cloud storage is of great concern in the academics [9], the industry [20] and the government [25]. This problem can be overcome by enciphering the data before storing it in cloud storage and retrieving it by deciphering. However, especially commercial users use conventional encryption algorithm [3, 13, 23, 32, 41, 46] such as AES, DES, Blowfish, etc. to encrypt their confidential data before storing it in cloud storage. But the time complexity, storage space and cryptographic computation are enormous in these conventional algorithms. The specific security requirements in cloud storage are largely cloudy to the end users. The two main security threats are sensitivity and integrity of the data received from remote storage. In an earlier article we proposed Secured Document Sharing using Visual Cryptography (SD-

SUVC) technique for efficient document storage, which utilizes only less storage space and time complexity for the document retrieval and it provides data confidentiality [8] to some extent.

This paper proposes a novel method named Securing Portable Document Format file Using Extended Visual Cryptography (SPDFUEVC) for more efficient file storage with absolute data confidentiality and integrity. It requires only less storage space on cloud and less time complexity for the retrieval of original PDF file using this algorithm.

The remainder of this article is arranged as follows. Section 2 covers a study of related work, Section 3 describes the architecture of SPDFUEVC technique, Section 4 discusses encryption, Section 5 illustrates decryption, Section 6 analyses the experimental results, Section 7 covers computational complexity and Section 8 concludes the article.

2 Related Work

Visual Cryptography was invented by Naor and Shamir in 1994 at the Eurocrypt Conference. This new cryptographic method is perfectly secure which can encode and decode the secret image without any cryptographic computations [11, 23, 35]. This system divides the secret image into two shares such as cipher and key transparencies, which are indistinguishable from random noise. The original secret image is obtained by placing key transparency over the cipher transparency.

The basic model of visual cryptography uses binary image which consists of collection of black and white pixels handling each of them separately. The secret image is divided into 'n' shares and each pixel appears in 'n' shares. The resultant image can be described as 'n' out of 'm' Boolean matrix

$$S = s_{ij}$$

where, $S_{ij} = 1$, the j^{th} subpixel in the i^{th} share being black; $S_{ij} = 0$, the j^{th} subpixel in the i^{th} share being white. Combine the shares s_1, s_2, \dots, s_n which properly aligns the sub pixels to get the original image.

To illustrate the concept of Visual Cryptography, the simplest version of two out of two scheme, where each original pixel of the secret image is coded into a pair of subpixels in each of the 2 shares specified in Figure 1. The drawbacks of this basic model are the huge size (the retrieved secret images two times larger than the original image) and the poor quality of the image [29].

The basic two out of two' visual cryptography techniques can be extended to $k \times n$ schemes [5, 7, 16, 17, 29, 44]. A more general model for visual cryptography based on general access structure and authorized and forbidden subsets of the participants has been developed. This model reduces the pixel expansion but it produces only optimal contrast of the image [4]. Hence the basic scheme in visual cryptography is restricted to binary image pattern which is insufficient in real time applications.















Pixel	Share-1	Share-2	Combination of two shares
	 	 	 
	 	 	 

Figure 1: Encoding black and white pixels

The pixel in grey level images ranges from 0 to 255. The limitation of this method is pixel expansion and low contrast of resultant image [6]. Instead of using grey pixels directly into the image shares, the grey level image is converted into binary image by a dithering technique. The resultant binary image is applied to traditional visual cryptography scheme. This scheme reduces the pixel expansion but produces only optimum quality of the image [22]. The grey scale image is further enhanced by using half-toning method which converts it into binary image and then visual cryptography scheme is applied to resultant image. But this scheme is not suitable for a large sized secret image although there is an enhancement in contrast when compared with the previous case [26]. There is an enhancement only in visual cryptography being directly applied to grey scale image but the limitation of this scheme is low contrast of the image [26].

Most of the real time information contains color images and Visual Cryptographic algorithm is applied on them for securing the original information more effectively [18, 33]. Though this method reduces the pixel expansion it obtains only optimum quality of the image [2, 15].

Standard Visual Cryptography algorithm creates noisy pixel on image shares which shows that some secret information is embedded in them. This issue can be overcome by applying the Extended Visual Cryptography algorithm. The secret information hidden in these cover images cannot be easily identified by anyone other than the owner of the file [42]. This scheme is further enhanced by meaningful shares being generated by dithering technique [34].

All these previous schemes have dealt with sharing of merely one secret. Despite the merit of this scheme is its ability to hide more than one secret within a set of secrets, it has the limitation of large size and poor quality of the image [12, 14]. This scheme is further enhanced by sharing multiple secrets without pixel expansion and good quality of the image [17, 28, 30]. So far all the current visual cryptographic algorithms have been applied exclusively only on images and not on pdf files.

3 SPDFUEVC Architecture

SPDFUEVC (Securing Portable Document Format file Using Visual Cryptography) technique mainly uses Extended Visual Cryptography to protect a secret pdf file in cloud storage. In all the current work in the domain of cloud computing, security is focused on using conventional encryption algorithm AES (Advanced Encryption Standard) for storing and retrieving data [34, 35]. This traditional encryption technique requires more time, space and also involves complex computations. Therefore the proposed SPDFUEVC technique can effectively replace the use of conventional encryption algorithm by using the Extended Visual Cryptography for uploading and downloading secret data. The overall concept of the system is very simple and it also protects the secret in the pdf file. For the security purpose, instead of uploading the original pdf file, it must be converted into a text file, next into image shares and then the resultant image shares, transformed into scrambled images and finally uploaded in to the cloud. Later the random noisy image shares should be extracted from the downloaded scrambled images and converted into the text file and again into the original pdf file. The following process is to be followed for uploading a pdf file into a cloud and downloading a pdf file from the cloud as shown in Figure 2.

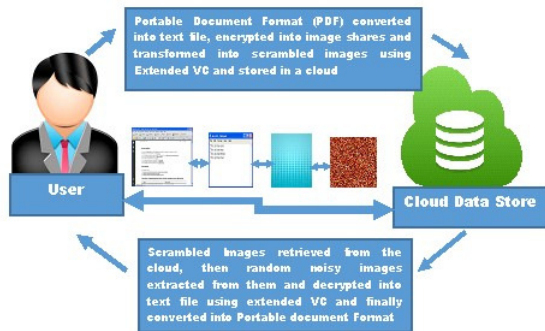


Figure 2: SPDFUEVC architecture

Algorithm 1 Uploading a secret pdf file

- 1: Select the pdf file which is to be uploaded
 - 2: Convert the pdf file into text file
 - 3: Convert the text file into image files and convert the resultant image files into scrambled image files using SPDFUEVC technique
 - 4: Upload the scrambled image files on a cloud
-

4 SPDFUEVC Encryption Process

When a pdf file which contains some valuable data is to be uploaded, it has to be encrypted with SPDFUEVC

Algorithm 2 Downloading the Secret pdf file

- 1: Select the scrambled image files which are to be downloaded
 - 2: Download the scrambled image files from the cloud
 - 3: Extract image files from the scrambled image files
 - 4: Obtain the resultant text file by stacking the image files using SPDFUEVC technique
 - 5: Convert the text file into the original pdf file
-

encryption algorithm which involves three phases. In the initial phase, the pdf file must be converted into a text file using Apache PDFbox application programming interface and in the second phase the resultant text file must be encrypted using SPDFUEVC encryption technique. Each character in every line in the text file must be taken up and converted into an integer (ascii value). Then it is transformed into a pixel using SetRGB method and fed on a buffered image. Every pixel in a line must be stored alternately one in the first image and the next one in another image. This should be continued till the end of the file. Finally the image shares are converted into scrambled images and the scrambled images must be uploaded into a cloud as shown in Figure 3.

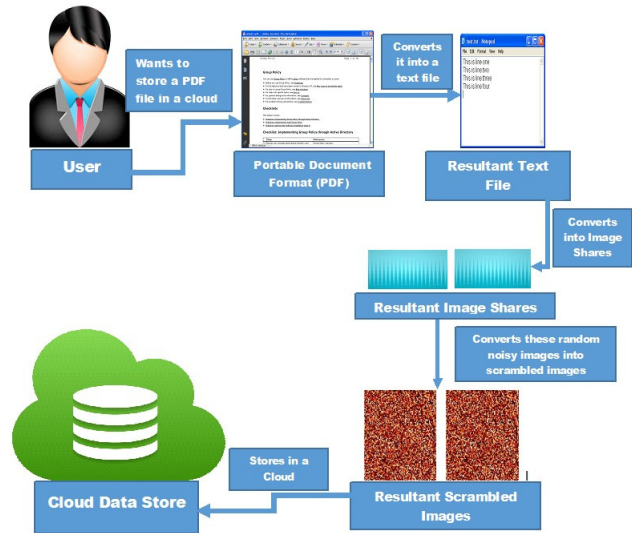


Figure 3: Secret pdf file in cloud storage

Algorithm 3 Conversion of original pdf file into text file

- 1: **Input:** pdf file
 - 2: **Output:** Text File
 - 3: Use the relevant PDFBox package to convert pdf file into text file
 - 4: Read the pdf file such as Rama.pdf
 - 5: PDFParser is used to extract the text from the parser
 - 6: Use getText() function to retrieve all the strings from the pdf file
 - 7: Store the extracted strings into text file
-

Algorithm 4 Algorithm for encryption of text file into image shares

- 1: **Input:** Text file
- 2: **Output:** Image Shares
- 3: Read the text file which contains some secret information
- 4: Initialize the 2 random noisy image shares such as Image1, Image2 with png format
- 5: Move all the information from text file into buffer
- 6: Calculate the height and width of the image shares
- 7: Width = No. of Characters in a line
- 8: Height = No. of lines in a file
- 9: Read each line from the contents in a buffer
- 10: begin
- 11: Select each character from the line
- 12: Compute ascii value for that character
- 13: Calculate the individual pixel using SetRGB() in Java
- 14: Place every pixel of a line in Image1 and Image2 alternately
- 15: Store pixel on the image shares based on its x, y coordinates which denote the position of the character in a line and line number respectively
- 16: This process is continued till the end of the file
- 17: end
- 18: Finally save the Image shares such as Image1, Image2 in png format using ImageIo.write () in Java

Algorithm 5 Algorithm for converting random noisy images into scrambled images

- 1: **Input:** Image files
- 2: **Output:** Scrambled Images
- 3: Initialize the two scrambled image shares as Image1, Image2 with png format
- 4: Interchange width and height of the random noisy images as height and width of the scrambled images
- 5: Read the pixel in each random noisy image share
- 6: begin
- 7: Get the pixel value using getRGB method
- 8: Store that pixel in the corresponding scrambled image using setRGB method
- 9: This process is repeated till the end of the file
- 10: end
- 11: Finally save the scrambled Image shares such as Image1, Image2 in png format

5 SPDFUEVC Decryption Process

Retrieving the original pdf file from the scrambled images stored in the cloud storage using SPDFUEVC decryption technique involves three phases. In the initial phase, the random noisy image shares are extracted from the scrambled images and then they are converted into a text file in the next phase. The image shares are in.png format and every line of them must be read. Then each pixel from every line must be retrieved using getRGB method.

Each one of them must be rewritten in hexa code and the resultant value entered in a string buffer. This should be continued till the end of the file. Finally the entire contents in the buffer must be rewritten as a text file and in the last phase, the text file is converted into the original pdf using Apache PDFbox application program interface as shown in Figure 4.

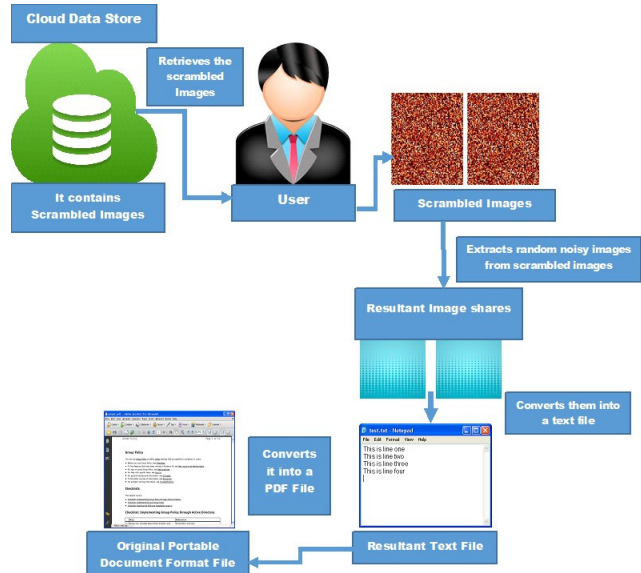


Figure 4: PDF file retrieved from scrambled images

Algorithm 6 Extraction of random noisy images from the scrambled images

- 1: **Input:** Scrambled images
- 2: **Output:** Random noisy images
- 3: Initialize the two random noisy image shares as Image1, Image2 with png format
- 4: Interchange width and height of the scrambled images as height and width of the random noisy images
- 5: Read the pixel in each scrambled image share
- 6: begin
- 7: Get the pixel value using getRGB method
- 8: Store that pixel in the corresponding random noisy image using setRGB method
- 9: This process is repeated till the end of the file
- 10: end
- 11: Finally save the random noisy image shares such as Image1, Image2 in png format

6 Security Analysis

In this section, the security attribute of this technique have been discussed with various factors.

Confidentiality. The proposed technique reveals only the encrypted file information to the CSP and all

Algorithm 7 Algorithm for encryption of Text file into Image shares

- 1: **Input:** Random noisy image files
- 2: **Output:** Secret text file
- 3: Read the random noisy image files
- 4: Initialize the string buffer
- 5: Select each and every pixel from the line using getRGB() in Java
- 6: Find ascii value for the selected pixel
- 7: Find appropriate character from ascii value and place it in a buffer
- 8: Rewrite all the data from the buffer into a text file using FileWriter() in Java
- 9: Finally save the Image files in png format

Algorithm 8 Conversion of a text file into original pdf file

- 1: **Input:** Text file
- 2: **Output:** pdf file
- 3: Use file reader to read the source file
- 4: Store all the contents to buffered image
- 5: Create a pdf file using PDFwriter component
- 6: Use pdfDoc.setMargins() function to set the margins of pdf file
- 7: Read each and every line from the buffer
- 8: Set the font size and style with setFontFamily(), setFontSize() function respectively
- 9: Store the text in the document using setText() function
- 10: Repeat the process until file comes to an end
- 11: Finally save the content as a pdf file

other details maintained by TTP (Trusted third party). This ensure that sensitive information protected from CSP and illegal user.

Integrity. The original data files are converted into scrambled random noisy shares and then uploaded into the cloud. The proposed technique ensure that no one can modify the data.

Access Control Management. The DO is revealing the file access control details to the TTP in a secure manner. This avoids the illegal users to modify the access control details.

Prevention of Intruder. The proposed technique prevents the intruder to access the data transferred in the communication channel between the communicating parties such as DO, CSP, User and TTP by enforcing the encrypted form of data transfer.

7 Experimental Result

The above-said SPDFUEVC technique has been evaluated in Java [19, 31] and various tests have been worked out using a laptop with the configuration of 2.40 GHz,

Intel core i3 processors with 4GB RAM on Windows 8 Professional version 1. The algorithm is implemented in various sizes of pdf files; the performance of the technique is evaluated with parameters such as execution time and size of the image shares for SPDFUEVC encryption and decryption technique. During the encryption process the original pdf file is converted into a text file then it is converted into random noisy image shares which are then converted into scrambled images. The data confidentiality of this algorithm is compared to that of conventional symmetric encryption algorithms such as AES (Advanced Encryption Standard) and DES (Data Encryption Standard) used in the current cloud domain [1, 40].

The test result of the pdf file for the proposed SPDFUEVC technique is as follows.

7.1 SPDFUEVC Encryption Process

7.1.1 Conversion of the pdf File into a Text File

The pdf file is converted into a text file using apache PDF Box program interface. Figure 5 and Figure 6 show the sample pdf file and the resultant text file after the conversion.

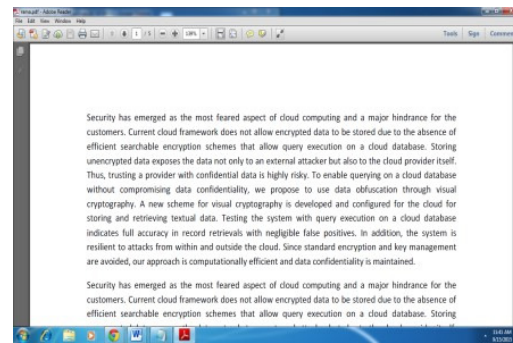


Figure 5: Sample pdf file

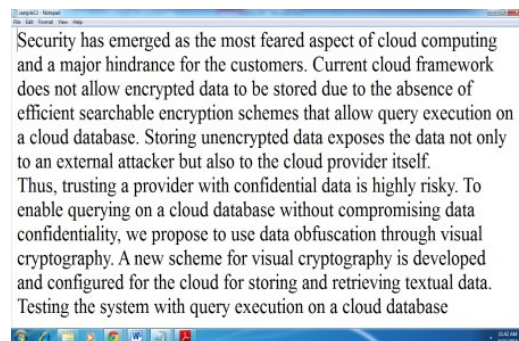


Figure 6: Resultant text file

7.1.2 Conversion of Text File into Image Shares Using SPDFUEVC Encryption Technique

The resultant text file is encrypted into image files using SPDFUEVC encryption technique. Figures 7 and 8 show

the resultant image files after encryption process.

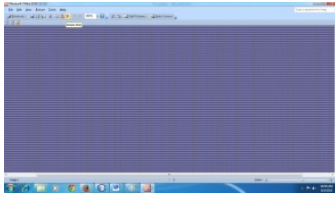


Figure 7: Image1.png (1090 x 10)

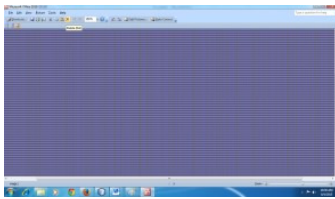


Figure 8: Image2.png (1090 x 10)

7.1.3 Algorithm for Converting Random Noisy Image Shares into Scrambled Images

The resultant image files are converted into scrambled images. Figures 9 and 10 show the random noisy images converted into scrambled images.



Figure 9: Scrambled Image1.png (10 x 1090)



Figure 10: Scrambled Image2.png (10 x 1090)

7.2 SPDFUEVC Decryption Process

7.2.1 Extraction of Random Noisy Images from the Scrambled Images

The random noisy image shares are extracted from scrambled images. Figure 11 and Figure 12 show the random noisy images extracted from scrambled images.

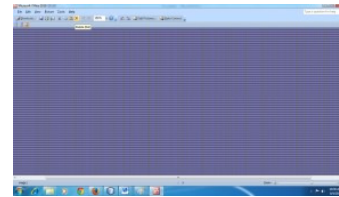


Figure 11: Image1.png (1090 x 10)

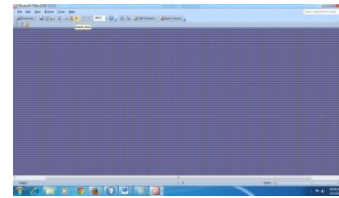


Figure 12: Image2.png (1090 x 10)

7.2.2 Conversion into Text File from the Random Noisy Image Shares

The random noisy image shares are converted into a text file using SPDFUEVC technique. Figure 13 shows the resultant text file after the decryption process.

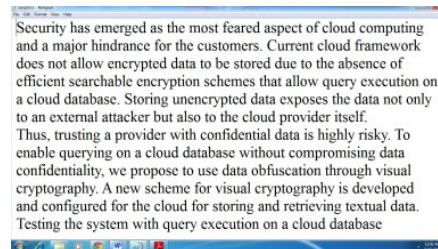


Figure 13: Resultant text file

7.2.3 Conversion of Text File into pdf File

The resultant text file is converted into pdf file using apache PDF Box application programming interface. Figure 14 shows the resultant pdf file after the conversion.

This algorithm has been applied on various pdf files and it is found that the size of the image file is comparatively lesser than that of the original pdf file during the SPDFUEVC encryption algorithm. Table 1 describes the different sizes of pdf files juxtaposed with those of image files and Figure 15 juxtaposes the sizes of the original pdf files with those of image files.

Table 2 juxtaposes the sizes of the original pdf files with those of the deciphered pdf files and Figure 16 discusses the correlation of the sizes of the original pdf files with those of the deciphered pdf files. This analysis proves that both the pdf files are of same size.

7.2.4 Execution Time for Encryption

The time taken for conversion of pdf file into text file, then into image shares and then into scrambled images

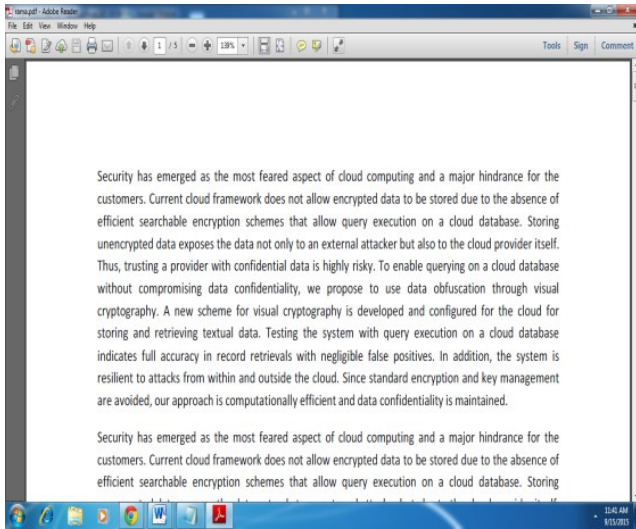


Figure 14: Resultant pdf file

Table 1: Size of Image shares during SPDFUEVC encryption

Size of original pdf files (in KB)	Size of image files (in KB)
6	2
9	2
15	4
21	4
30	4
38	4

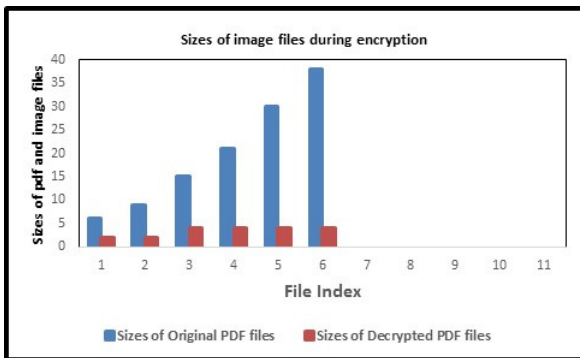


Figure 15: Size of image files during SPDFUEVC encryption

is calculated as execution time in encryption process and also the time for conversion of scrambled images into random noisy image shares, then into text file and finally into original pdf file is considered as execution time in decryption process. This process is compared with conventional DES and AES algorithms which are used in the current cloud domain. Table 3 compares this execution time with that of AES and DES.

The execution time for encryption in SPDFUEVC

Table 2: Comparison of sizes of original and decrypted pdf files

Size of original pdf files (in KB)	Size of deciphered pdf files (in KB)
6	6
9	9
15	15
21	21
30	30
38	38

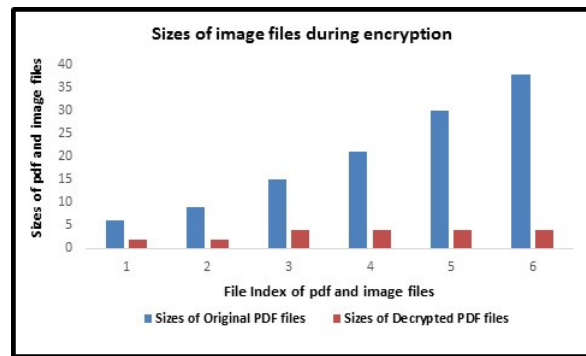


Figure 16: Size of image files during SPDFUEVC decryption

Table 3: Execution time for encryption in SPDFUEVC, AES and DES

Size of Pdf files (in Kb)	Execution time of SPDFUEVC (in ms)	Execution time for AES (in ms)	Execution time for DES(in ms)
6	431	671	531
9	483	702	565
15	655	862	734
21	734	934	804
30	890	1090	950
38	1077	1251	1111

technique is found to be comparatively lesser than those in DES and AES for various sizes of pdf files. The Figure 17 depicts the time taken for the execution of these algorithms.

7.3 Execution Time for Decryption

Table 4 compares the execution time for decryption in SPDFUEVC with those of AES and DES. The execution time for decryption in SPDFUEVC technique is found to be marginally lesser than those in DES and AES. The Figure 18 denotes the time taken for the execution of these algorithms.

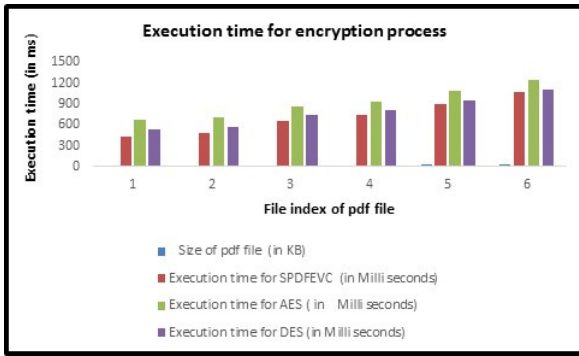


Figure 17: Encryption execution time for SPDFUEVC, AES and DES

Table 4: Execution time for decryption in SPDFUEVC, AES and DES

Size of Pdf files (in Kb)	Execution time of SPDFUEVC (in ms)	Execution time for AES (in ms)	Execution time for DES(in ms)
6	277	282	280
9	284	288	286
15	285	293	289
21	302	306	304
30	309	314	312
38	316	320	318

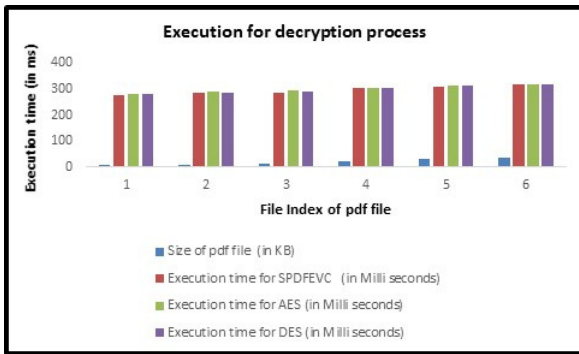


Figure 18: Decryption execution time for SPDFUEVC, AES and DES

8 Complexity

All the previous visual cryptography approaches have been so far used to hide only a small amount of information in an image. In the current scenario, when researchers try to hide a large amount of information in it, they have to face the challenges of increased share size and image processing time and also poor quality of retrieved image resulting in obtaining only optimum solutions [12, 38, 47]. The proposed SPDFUEVC technique can effectively hide a larger amount of textual data with minimum effort, space and time complexity and retrieve the whole information with data confidentiality and in-

tegrity.

9 Conclusion

The proposed new technique named (SPDFUEVC) Securing Portable Document Format file Using Extended Visual Cryptography ensures data integrity and confidentiality in the cloud storage. The traditional visual cryptography technique has so far assured confidentiality only to image file. But the proposed approach provides the same for the pdf file using visual cryptography technique and proves to be more efficient than the current cloud storage techniques. The complexity of this approach is shown to be reasonable and it is much less than those of standard algorithms. In this technique, storage entry is fully protected and hence prohibitive to any unauthorized entity. The proposed fool-proof technique ensures data confidentiality and security along with integrity and reputation.

References

- [1] Amazon, *Amazon EBS Encryption Now Available*, May 2014. (<https://aws.amazon.com/about-aws/whats-new/2014/05/21/Amazon-EBS-encryption-now-available/>)
- [2] S. Abdulla, "New visual cryptography algorithm for colored image," *Journal of Computing*, vol. 2, no. 4, pp. 4–15, 2010.
- [3] M. Arunachalam and K. Subramanian, "Aes based multimodal biometric authentication using cryptographic level fusion with fingerprint and finger knuckle print," *International Arab Journal of Information Technology*, vol. 12, no. 5, pp. 431–440, 2015.
- [4] G. Ateniese, C. Blundo, A. De Santis, and D. R. Stinson, "Visual cryptography for general access structures," *Information and Computation*, vol. 129, no. 2, pp. 86–106, 1996.
- [5] C. Blundo, P. D'Arco, A. De Santis, and D. R. Stinson, "Contrast optimal threshold visual cryptography schemes," *SIAM Journal on Discrete Mathematics*, vol. 16, no. 2, pp. 224–261, 2003.
- [6] C. Blundo, A. De Santis, and M. Naor, "Visual cryptography for grey level images," *Information Processing Letters*, vol. 75, no. 6, pp. 255–259, 2000.
- [7] C. Blundo, A. De Santis, and D. R. Stinson, "On the contrast in visual cryptography schemes," *Journal of Cryptology*, vol. 12, no. 4, pp. 261–289, 1999.
- [8] K. Brindha and N. Jeyanthi, "Secured document sharing using visual cryptography in cloud data storage," *Cybernetics and Information Technologies*, vol. 15, no. 4, pp. 111–123.
- [9] R. Buyya, C. S. Yeo, S. Venugopal, J. Broberg, and I. Brandic, "Cloud computing and emerging it platforms: Vision, hype, and reality for delivering computing as the 5th utility," *Future Generation Computer Systems*, vol. 25, no. 6, pp. 599–616, 2009.

- [10] Z. Cao, C. Mao, 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.
- [11] V. Chang and M. Ramachandran, "Towards achieving data security with the cloud computing adoption framework," *IEEE Transactions on Services Computing*, vol. 9, no. 1, pp. 138–151, 2016.
- [12] S. Chen, "A visual cryptography based system for sharing multiple secret images," in *Proceedings of the 7th WSEAS International Conference on Signal Processing*, pp. 113–118, Chicago, Aug. 2007.
- [13] C. Esposito, A. Castiglione, and K. K. R. Choo, "Encryption-based solution for data sovereignty in federated clouds," *IEEE Cloud Computing*, vol. 3, no. 1, pp. 12–17, 2016.
- [14] J. B. Feng, H. C. Wu, C. S. Tsai, Y. F. Chang, and Y. P. Chu, "Visual secret sharing for multiple secrets," *Pattern Recognition*, vol. 41, no. 12, pp. 3572–3581, 2008.
- [15] B. L. Gunjal and S. N. Mali, "Design and implementation of invisible and visible color image watermarking with netbeans ide," *International Journal of Computer Applications*, vol. 71, no. 11, pp. 25–45, 2013.
- [16] T. Hofmeister, M. Krause, and H. U. Simon, "Contrast-optimal k out of n secret sharing schemes in visual cryptography," *Theoretical Computer Science*, vol. 240, no. 2, pp. 471–485, 2000.
- [17] A. E. A. El Hossaini, M. El Aroussi, K. Jamali, S. Mbarki, and M. Wahbi, "A new robust blind copyright protection scheme based on visual cryptography and steerable pyramid," *International Journal of Network Security*, vol. 18, no. 2, pp. 250–262, 2016.
- [18] Y. C. Hou, "Visual cryptography for color images," *Pattern Recognition*, vol. 36, no. 7, pp. 1619–1629, 2003.
- [19] Java, *Java*, 2015. (<http://www.java2s.com/>)
- [20] L. M. Kaufman, "Data security in the world of cloud computing," *IEEE Security & Privacy*, vol. 7, no. 4, pp. 61–64, 2009.
- [21] A. Le, A. Markopoulou, and A. G. Dimakis, "Auditing for distributed storage systems," *IEEE/ACM Transactions on Networking*, vol. 24, no. 4, pp. 2182–2195, 2016.
- [22] C. C. Lin and W. H. Tsai, "Visual cryptography for gray-level images by dithering techniques," *Pattern Recognition Letters*, vol. 24, no. 1, pp. 349–358, 2003.
- [23] C. H. Ling, C. C. Lee, C. C. Yang, and M. S. Hwang, "A secure and efficient one-time password authentication scheme for WSN," *International Journal of Network Security*, vol. 19, no. 2, pp. 177–181, 2017.
- [24] C. Liu, R. Ranjan, X. Zhang, C. Yang, D. Georgakopoulos, and J. Chen, "Public auditing for big data storage in cloud computing—a survey," in *IEEE 16th International Conference on Computational Science and Engineering*, pp. 1128–1135, harvard, Dec. 2013.
- [25] P. Mell and T. Grance, "Effectively and securely using the cloud computing paradigm," tech. rep., Oct. 2009.
- [26] A. Kr. Mishra and A. Gupta, "Visual cryptography for gray scale image using block replacement half toning method," *African Journal of Computing & ICT*, vol. 6, no. 4, pp. 53–58, 2013.
- [27] 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.
- [28] M. Naor and A. Shamir, "Visual cryptography," in *Workshop on the Theory and Application of Cryptographic Techniques*, pp. 1–12, berlin, May 1994.
- [29] M. Naor and A. Shamir, "Visual cryptography ii: Improving the contrast via the cover base," in *International Workshop on Security Protocols*, pp. 197–202, berlin, apr. 1996.
- [30] P. K. Naskar, H. N. Khan, and A. Chaudhuri, "A key based secure threshold cryptography for secret image," *International Journal of Network Security*, vol. 18, no. 1, pp. 68–81, 2016.
- [31] Netbeans, *Netbeans IDE*, 2012. (<https://netbeans.org/>)
- [32] N. Ojha and S. Padhye, "Cryptanalysis of multi prime rsa with secret key greater than public key," *International Journal of Network Security*, vol. 16, no. 1, pp. 53–57, 2014.
- [33] R. De Prisco and A. De Santis, "Color visual cryptography schemes for black and white secret images," *Theoretical Computer Science*, vol. 5, no. 10, pp. 62–86, 2013.
- [34] J. Sandeep and A. Manjeed, "Embedded extended visual cryptography scheme," *Journal of Computer Engineering*, vol. 8, no. 1, pp. 41–47, 2012.
- [35] C. E. Shannon, "A mathematical theory of communication," *Mobile Computing and Communications Review*, vol. 5, no. 1, pp. 3–55, 2001.
- [36] J. Singh, "Cyber-attacks in cloud computing: A case study," *International Journal of Electronics and Information Engineering*, vol. 1, no. 2, pp. 78–87, 2014.
- [37] W. Teng, G. Yang, Y. Xiang, T. Zhang, and D. Wang, "Attribute-based access control with constant-size ciphertext in cloud computing," *IEEE Transactions on Cloud Computing*, vol. pp, no. 99, pp. 1–1, 2015.
- [38] C. C. Thien and J. C. Lin, "Secret image sharing," *Computers & Graphics*, vol. 26, no. 5, pp. 765–770, 2002.
- [39] H. Tian, Y. Chen, C. C. Chang, H. Jiang, Y. Huang, Y. Chen, and J. Liu, "Dynamic-hash-table based public auditing for secure cloud storage," *IEEE Transactions on Services Computing*, 2015. (doi: 10.1109/TSC.2015.2512589)
- [40] Z. Wan, J. E. Liu, and R. H. Deng, "Hasbe: a hierarchical attribute-based solution for flexible and scalable access control in cloud computing," *IEEE*

Transactions on Information Forensics and Security, vol. 7, no. 2, pp. 743–754, 2012.

- [41] B. Wang, M. Li, and H. Wang, “Geometric range search on encrypted spatial data,” *IEEE Transactions on Information Forensics and Security*, vol. 11, no. 4, pp. 704–719, 2016.
- [42] D. Wang, F. Yi, and X. Li, “On general construction for extended visual cryptography schemes,” *Pattern Recognition*, vol. 42, no. 11, pp. 3071–3082, 2009.
- [43] Z. Wang, Y. Lu, G. Sun, “A policy-based deduplication mechanism for securing cloud storage,” *International Journal of Electronics and Information Engineering*, vol. 2, no. 2, pp. 70–79, 2015.
- [44] C. Wu and L. Chen, “A study on visual cryptography (masters thesis),” tech. rep., Sept. 1998.
- [45] Z. Yan, W. Ding, X. u, H. Zhu, and R. H. Deng, “Deduplication on encrypted big data in cloud,” *IEEE Transactions on Big Data*, vol. 2, no. 2, pp. 138–150, 2016.
- [46] Z. Yan, M. Wang, Y. Li, and A. V. Vasilakos, “Encrypted data management with deduplication in cloud computing,” *IEEE Cloud Computing*, vol. 3, no. 2, pp. 28–35, 2016.
- [47] C. N. Yang and T. S. Chen, “Size-adjustable visual secret sharing schemes,” *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences*, vol. 88, no. 9, pp. 2471–2474, 2005.

Biography

K. Brindha, M. E.(CSE) from Sathayabama University, Chennai, Tamilnadu, India is presently working as Assistant Professor (Selection Grade) in the School of Information Technology and Engineering, VIT University, Vellore. Currently she is doing research work at VIT University, Vellore, Tamil nadu, India. She is the author of several articles published in reputed journals. Her research interest include cryptography, network security and image processing.

Dr. N. Jeyanthi is an Associate Professor in School of Information Technology and Engineering, VIT University, India. She received her Ph.D. and M.Tech. in Information Technology with Networking as specialisation from VIT University, BE in Computer Science and Engineering from Madurai Kamaraj University, India. Her research interest is on network security in real-time applications. She published around thirty international journal papers and many international conference papers. She received the Active Researcher Award from VIT University for three consecutive years. She is an editorial board member of international journals and acted as programme chair in many international conferences. She is a life member of Indian Society of Technical Education. Her research interest includes Cryptography, Network Security, IoT.

Design and Implementation of Secure Remote e-Voting System Using Homomorphic Encryption

Ihsan Jabbar and Saad Najim Alsaad

(Corresponding author: Ihsan Jabbar)

Department of Computer Science, College of Science, University of Mustansiriyah

Safi Al Din Al Hilli Street, Baghdad, Iraq

(Email: ihsan.jabbar90@gmail.com)

(Received Jan. 28, 2016; revised and accepted Apr. 23 & Oct. 25, 2016)

Abstract

Internet polling also known as “e-voting” became popular in past few years, since it reduces the tallying cost and time, increases the number of voter participation, also reduces the human resources and the traditional work that means less fraud and corruption. In this paper, a remote e-Voting system is designed and implemented using homomorphic encryption. The homomorphic property in ElGamal cryptosystem are exploited to achieve two important voting requirements: first, the security of device used for electronic voting by voter. Second, the voter has the ability to choice willfully and uncoercionly. The general voting system requirements such as eligibility, privacy, accuracy, fairness, Receipt-freeness, coercion resistance, mobility, simplicity, individual verifiability, scalability and availability are also achieved in the system.

Keywords: Electronic Voting; ElGamal; Homomorphic Encryption

1 Introduction

Electronic voting is a process completely conducted by electronic devices such as computers and communication technologies. Its applications such as elections are so sensitive in terms of security. Current e-Voting schemes are based on either mix network, or blind signatures, or homomorphic encryption. Homomorphic encryption is used to make sure that votes holds its confidentiality by encrypting and calculating all votes without decrypting.

Voting schemes based on homomorphic encryption were first introduced by Benaloh [5]. Several improved schemes were developed after that. These schemes follow the similar election procedures, but they introduce new security properties, such as receipt-freeness. In 1997, Crammer et al. [7], introduced a new multi-authority secret-ballot election scheme based on the discrete log assumption. Their scheme achieves privacy, universal verifiability and robustness. In 2002, Rivest [18] discussed in

his lecture notes a voting scheme based on homomorphic property of Paillier cryptosystem to achieve the privacy of voters by tallying the encrypted votes. This scheme used blind signature which allow for anonymous voting. In 2010, George and Sebastian [10] presented a voting scheme based on homomorphic encryption. The scheme achieves privacy, uncoercibility, and receipt-freeness. The scheme can be used for both yes/no and multi-candidates types of voting. In 2011, Huszti [13] proposed a homomorphic encryption-based voting scheme based on Crammer Scheme [7]. The scheme achieves eligibility, unreusability, privacy, verifiability, receipt-freeness, and uncoercibility. It only needs anonymous channels. In 2013, Hussien and Aboelnaga [12] proposed a new voting scheme based on additive homomorphic property of Paillier cryptosystem and blind signature based on RSA. The scheme achieves eligibility, secrecy, uniqueness, privacy and accuracy. In 2013, Yi and Okamoto [22] presented voting scheme which maintains the privacy of voter even if the voter’s PC infected by malware or the voter is physically controlled by the adversary. The scheme can only tell if the candidate wins or loses without the number of yes or no votes. In 2014, Zhao et al. [23] presented a voting scheme based on homomorphic encryption to ensure anonymity, privacy and reliability. The scheme using RSA cryptosystem to encrypted the data. In 2015, Will et al. [21] described a partially homomorphic cloud-based mobile voting system. They implemented the system to show its practicality. The system achieves eligibility, unreusability, untraceability, verifiability, tally correctness, uncoerceability, auditability, accessibility, fairness, soundness and integrity.

In this paper, a secure e-Voting system based on homomorphic property of ElGamal cryptosystem is designed and implemented. Our system achieved the following e-Voting system requirements: eligibility, privacy, accuracy, fairness, receipt-freeness, coercion resistance, mobility, simplicity, individual verifiability, scalability and availability.

The rest of this paper is organized as follows: Section 2 provides the background of homomorphic encryption

tion. Section 3 presents ElGamal cryptosystem. The design and implementation of the system are introduced in Section 4. In Section 5, a very simple testing example is given. The security analysis is discussed in Section 6. Finally, our conclusions are drawn in Section 7.

2 Homomorphic Encryption

Homomorphic encryption is a type of encryption that allows particular computations to be conducted on ciphertext and return an encrypted result, the decrypted of result is equal the result of conducting the operation on the plaintext. The property of homomorphic is useful to develop a secure e-voting system with high privacy data retrieving scheme, also it makes the use of cloud computing by ensuring the privacy of processed data. An example for its mathematical consistency, if there are two numbers 10 and 20 then both are encrypted to 56 and 69 respectively, the addition operator gives a number with value 125, the decrypted of this value is 30 [14].

The concept of homomorphic encryption was suggested in 1978 by Rivest and Adleman [19], But for 30 years the progress is very slow. In 1982, Goldwasser and Micali [11] proposed their encryption system that was able to encrypt one bit in additive homomorphic encryption. Paillier [16] in 1999 suggested another additive homomorphic encryption. Boneh, Goh and Kobi [2] in 2005 were invented a security system of encryption which conduct only single multiplication but large number of additions. In 2009, Gentry [9] construct a fully homomorphic encryption based system that able to conduct both of addition and multiplication, but the scheme is impractical. Several optimizations and refinements were proposed after that, but this schemes are still inefficient and impractical [20].

3 ElGamal Cryptosystem

Based on the Diffie-Hellman key exchange, Taher ElGamal [8] presents his public key cryptosystem in 1985. The security of ElGamal encryption scheme depends upon the difficulty of computing discrete logarithms over finite field. ElGamal scheme can be defined over any cyclic group G with a large prime order q and a generator g . The three components that configure the scheme are as shown in Figure 1.

ElGamal cryptosystem has a homomorphic property as follows [6]:

$$(C_{1,1}, C_{1,2}) = (g^{r_1}, p_1 \cdot y^{r_1})$$

and

$$(C_{2,1}, C_{2,2}) = (g^{r_2}, p_2 \cdot y^{r_2})$$

where r_1 and r_2 are randomly chosen from $\{1, 2, \dots, q-1\}$

<ul style="list-style-type: none"> • Key Generation
Select a large prime as a q Select x to be a member of the group $G = \langle Zq^*, X \rangle$, x must be " $1 \leq x \leq q-1$ " Select g to be a primitive root (generator) in the group $G = \langle Zq^*, X \rangle$ $y = g^x \text{ mod } q$ Public key $\leftarrow (g, y, q)$ Private key $\leftarrow x$
<ul style="list-style-type: none"> • Encryption
Select a random integer r in the group $G = \langle Zq^*, X \rangle$, r must be " $1 \leq r \leq q-1$ " $C_1 = g^r \text{ mod } q$ $C_2 = (p \cdot y^r) \text{ mod } q$ // p is the plaintext
<ul style="list-style-type: none"> • Decryption
$P = [C_2(C_1^{-x})^{-1}] \text{ mod } q$

Figure 1: ElGamal cryptosystem pseudocode

and $m_1, m_2 \in G$ someone can compute:

$$\begin{aligned} E(p_1).E(p_2) &= (C_{1,1}, C_{1,2}).(C_{2,1}, C_{2,2}) \\ &= [g^{r_1}.g^{r_2}, (p_1.y^{r_1}).(p_2.y^{r_2})] \\ &= [g^{r_1+r_2}, (p_1.p_2).y^{r_1+r_2}] \\ &= E(p_1.p_2), \end{aligned}$$

where E symbolizes to the encryption process.

4 Design and Implementation

Developing an e-Voting system requires the collaboration of many participants with different background. In our system, there are five participating actors: Administrator, Registrars, Tally Authorities, Candidates and Voters. Table 1, summarizes the participating actors and their responsibilities. The design of e-Voting System depicted in Figure 2, consists of four stages: election setup, registration, voting and tallying. These stages are consecutively, that's mean no feedback between one stage to another. We assume that there is a Bulletin Board (**BB**), an insert only board readable by the public. This system supports multi-candidate elections which has n_C of candidates. Each voter V_i cast his vote for each candidate. This vote may be Yes or No, this is equivalent to 1 and -1, respectively.

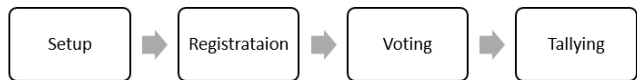


Figure 2: General structure of the system

- 1) **Election Setup:** The aim of this stage is to initialize election database and calculate necessary parameters to pass them to the next stages. The algorithm of this stage is depicted in Algorithm 1. It illustrates that two actors participate in this stage: Admin and Tally authority.

The implementation is based on ElGamal encryption scheme over a group G with a large prime order q and a generator g . These parameters should be determined by admin. In addition, the election database

Table 1: Actors participating and responsibilities

Actor	Responsibilities	Notes
<i>Administrator</i>	responsible for the eligible voters list, election setup and controlling the entire system	$N \setminus A$
<i>Registrars</i>	authorize the voters for the election during registration stage	Registrars are distributed on regions to facilitate the registration process
<i>Tally Authorities</i>	responsible for counting the votes and the announcement of the final results of the election	The list of tallying authorities $T = \{T_1, T_2, \dots, T_{n_T}\}$
<i>Candidates</i>	asking for the vote and competing with each other to get highest number of votes	The list of candidates $C = \{C_1, C_2, \dots, C_{n_C}\}$
<i>Voters</i>	qualified to vote and casting ballots	The list of voters $V = \{V_1, V_2, \dots, V_{n_V}\}$

Algorithm 1 Setup stage

Input: list of all eligible citizens to vote

Output: Election DB, $g, q, PVTKEY$ for T, n_T, n_C

- 1: Admin creates database and chooses g, q , and n_T
 - 2: **for all** i in T **do**
 - 3: Choose a prime t_i as private key ($PVTKEY_i$)
 - 4: Calculate $PUBKEY = g^{t_i}$
 - 5: **end for**
-

should be created by admin. It consists of five tables: administrator, registrars, tally authorities, candidates, and registered voters. Also, in this stage each tally authority T_i chooses a random prime t_i as a private key $PVTKEY_i$ from Z_q^* then calculates the public key using Equation (1)

$$PUBKEY_i = g^{t_i} \quad (1)$$

The sequence diagram of this stage is shown in Figure 3.

- 2) **Registration:** The aim of this stage is to enable eligible citizens to be registered for voting stage. The algorithm of this stage is depicted in Algorithm 2.

Algorithm 2 Registration stage

Input: Election DB, $g, q, PUBKEY$ for T, n_T, n_C
Output: $n_V, (A, B)$ for each v

- 1: **for all** v want to register **do**
 - 2: Registrar checks the eligibility of v
 - 3: Generate r for v
 - 4: Generate $y \in Z_q^*$
 - 5: $(A, B) = Enc(r)$
 - 6: Delete r and save ciphertexts (A, B)
 - 7: **end for**
-

In registration stage, remember that the number of candidates n_C is given. The voter V_i identify himself to the registration employee (registrar) by the identification card (citizen card). The registrar enters the required information into system to verify if the voter's information exists in the eligible voters list. If

the voter is eligible, the system sends a password to his provided email address. The voter V_i will proceed the process of registration in private booth. Each V_i can login into system with the password which was sent by the system to the voter's email address. The system generates a reference r_i for each V_i . This reference is an integer, it is a string of bits with length equal to the total number of candidates n_C . The system generates the reference using Equation (2).

$$r_i = a_{i,1} + a_{i,2} + \dots + a_{i,n_C} 2^{n_C-1}, \quad (2)$$

where $a_{i,j} \in \{0, 1\}$.

The reference r_i is encrypted using ElGamal cryptosystem into two ciphertexts using Equations (3) and (4).

$$A_{i,j} = g^{y_{i,j}} \quad (3)$$

$$B_{i,j} = \begin{cases} g^{(\prod_{t=1}^{n_T} PUBKEY_t)^{y_{i,j}}} & \text{if } a_{i,j} = 0 \\ g^{-1}(\prod_{t=1}^{n_T} PUBKEY_t)^{y_{i,j}} & \text{if } a_{i,j} = 1, \end{cases} \quad (4)$$

where $y_{i,j}$ is randomly chosen by the system from Z_q^* . Hence, the system permanently deletes the reference and keeps only its ciphertexts. Encrypted reference saved in a separately file until the elections day. At the end, the voter's information will be saved as new entries to the "registered voter" table in the database. The sequence diagram of this stage is shown in Figure 4.

- 3) **Voting:** The aim of this stage is to enable registered voters to cast their votes. In this stage, the system does not require a secret channel for the voters to cast their votes, also does not need to encrypt the votes, this allow the voters to verify that their votes are correctly included and not manipulated by malwares or viruses. The algorithm of this stage is depicted in Algorithm 3.

After the announcement of the candidates by Admin on Bulletin Board (**BB**), a voting stage is started. To cast his vote, the voter V_i should remember his

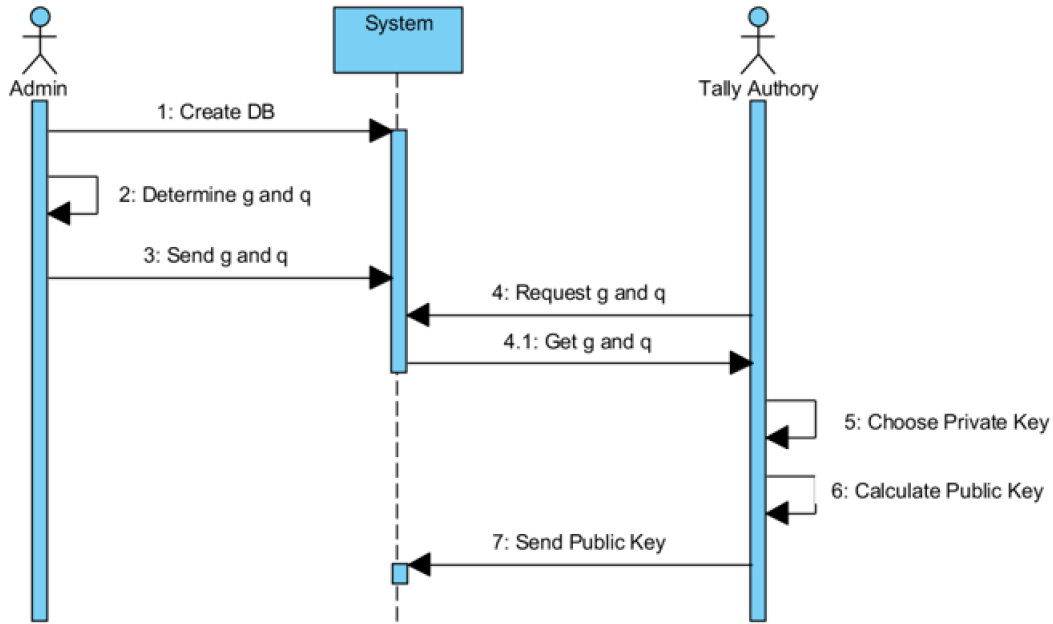


Figure 3: Election setup stage sequence diagram

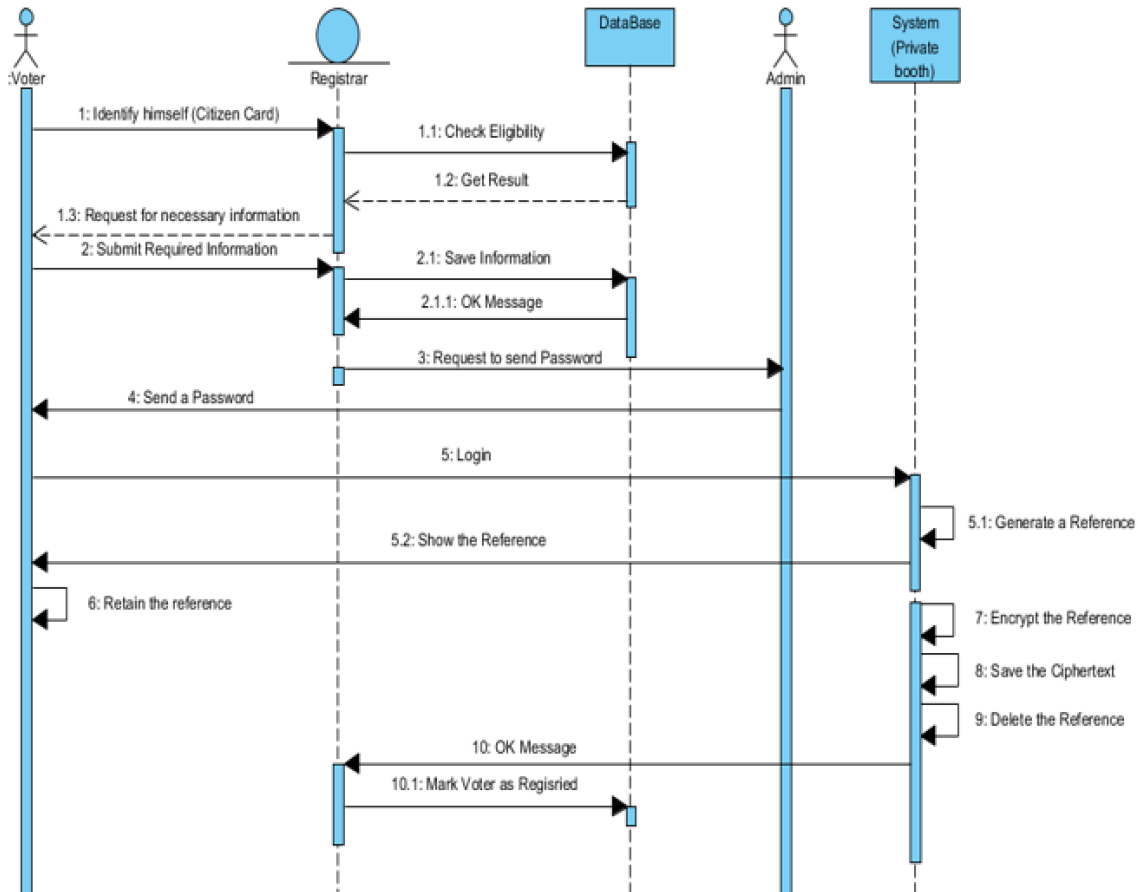


Figure 4: Registration stage sequence diagram

Algorithm 3 Voting stage

Input: Election DB, n_V, n_C
Output: β

- 1: Admin announces the list of candidates on **BB**
 - 2: **for all** i in V **do**
 - 3: **for all** j in C **do**
 - 4: V_i determines his $\beta_{i,j}$ and send it to the system
 - 5: Admin publishes $\beta_{i,j}$ on **BB**
 - 6: **end for**
 - 7: **end for**
-

reference r_i and login to the system with the required credentials. The voter V_i chooses a sequence of his $\beta_{i,j}$ with values of 1 or -1 according to the Table 2.

 Table 2: Voter guide to determine β

Reference	Vote	β
0	Yes	1
1	Yes	-1
0	No	-1
1	No	1

For example, if the number of candidates n_c is 4 and the reference number r_i is (0, 1, 1, 0). Suppose that, the voter V_i wants to select (Yes) for the third candidate and (No) for the other candidates, then the β_i should be (-1, 1, -1, -1). This is the point, the string of β_i dose not refers to what the voter is selected.

After entering β for each candidate, β sent to the system saved in separately and stand by for tallying. At the same time, the β will be published publicly on the (**BB**). The voters can know if their votes have been changed or not, everyone can access to the , but only the voter himself knows what does it means. The sequence diagram of this stage is shown in Figure 5.

- 4) **Tallying:** The aim of this stage is to count the ballots and get the final result for each candidate. Each tally authority should apply the algorithm (A) of this stage. This algorithm is depicted in Algorithm 4.

In this stage, tally authorities combine all valid ballots (β) posted on (**BB**) using the homomorphic property of ElGamal scheme as shown in Equations (5) and (6).

$$X_{T,j} = \prod_{i=1}^{n_V} A_{i,j}^{B_{i,j}} \quad (5)$$

$$Y_{T,j} = \prod_{i=1}^{n_V} B_{i,j}^{B_{i,j}}. \quad (6)$$

By using ElGamal encryption scheme, each tally authority T_i , calculates $X_{i,j}$ using Equation (7).

$$X_{i,j} = X_{T,j}^{PVTKEY_i}. \quad (7)$$

Algorithm 4 Algorithm (A) of tallying stage

Input: $n_T, n_C, n_V, PVTKEY, (A, B), \beta$
Output: X_{n_T} and Y_{n_T}

- 1: **for all** j in C **do**
 - 2: $X_{n_T} = 1, Y_{n_T} = 1$
 - 3: **for all** i in V **do**
 - 4: $X_{T,j} = X_{T,j} * A_{i,j}^{\beta_{i,j}}$
 - 5: $Y_{T,j} = Y_{T,j} * B_{i,j}^{\beta_{i,j}}$
 - 6: **end for**
 - 7: **end for**
 - 8: **for all** i in T **do**
 - 9: **for all** j in C **do**
 - 10: $X_{i,j} = X_{T,j}^{PVTKEY_i}$
 - 11: **end for**
 - 12: T_i sends $X_{i,j}$ and $Y_{T,j}$ to Admin
 - 13: **end for**
-

The algorithm (B) of tallying stage is applied by Admin as depicted in Algorithm 5.

Algorithm 5 Algorithm (B) of tallying stage

Input: Election DB, $g, n_T, n_C, n_V, X_{n_T}, Y_{n_T}$
Output: No. of (Yes/No) for each Candidate

- 1: **for all** j in C **do**
 - 2: $eq_j = 1$
 - 3: **for all** i in T **do**
 - 4: $eq_j = eq_j * X_{i,j}^{-1}$
 - 5: **end for**
 - 6: $g^{y_j - n_j} \leftarrow (eq_j * Y_{T,j})$
 - 7: $Z_j = \frac{\ln(g^{y_j - n_j})}{\ln(g)}$
 - 8: $y_j = \frac{Z_j + n_V}{2}$ // y_j : No. of (Yes) votes for C_j
 - 9: $n_j = n_V - y_j$ // n_j : No. of (No) votes for C_j
 - 10: Admin publishes y_j and n_j on **BB**
 - 11: **end for**
-

All X_{n_T} and Y_{n_T} are sent to admin. In turn, he calculates eq_j using Equation (8).

$$eq_j = Y_{T,j} \cdot \prod_{i=1}^{n_T} X_{i,j}^{-1} \quad (8)$$

eq_j is:

$$\begin{aligned} eq_j &= \prod_{i=1}^{n_T} g^{\beta_{i,j}(-1)^{a_{i,j}}} \\ &= \prod_{i=1}^{n_T} g^{v_{i,j}} \\ &= g^{y_j - n_j}, \end{aligned}$$

where y_j and n_j are represented the number of (Yes) and (No), respectively, for the candidate C_j . and:

$$y_j + n_j = n_V.$$

To determine the values of y_j and n_j using the fol-

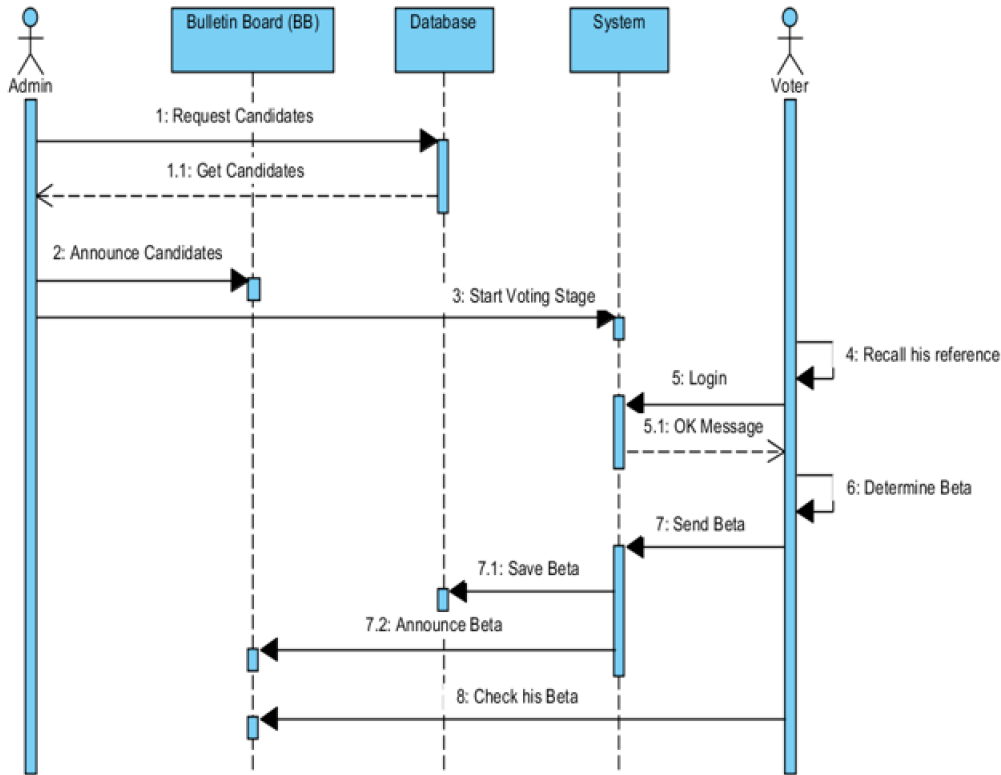


Figure 5: Voting stage sequence diagram

lowing equations:

$$g^{2y_j - n_V} = g^{y_j - n_j}$$

$$Z_j = \frac{\ln(g^{y_j - n_j})}{\ln(g)} \quad (9)$$

$$y_j = \frac{Z_j + n_V}{2} \quad (10)$$

$$n_j = n_V - y_j. \quad (11)$$

At the last, admin announce final results y_j and n_j for each candidate C_j on (BB). The sequence diagram of this stage is shown in Figure 6.

Figure 7, summarizes the four stages of the e-Voting system in more details focusing on the parameters of each stage.

5 Testing Example

Here is a simple example. We choose $q = 13$ and $g = 5$. The number of tally authorities $n_T = 3$. Table 3, illustrates the details of each tally authority.

The number of candidates $n_C = 3$, the number of voters $n_V = 10$. Suppose that, the references and votes of the voters as shown in Table 4. Table 4 also shows the corresponding β of these assumptions.

The voter V_7 in the table is taken to be an example of this testing. the reference of V_7 is: $r_7 = (0, 1, 0)$.

Table 3: A simple example (private key & public key for each tally authority)

i	T_i	$PVTKEY_i$	$PUBKEY_i$
1	T_1	7	78125
2	T_2	11	48828125
3	T_3	13	1220703125

The following results are the values of cipher texts A_7 and B_7 after encrypt r_i by applying Equations (3) and (4).

$$A_{7,1} = 125$$

$$B_{7,1} = 504870979341447555463506281780983186990852118469774723052978515625$$

$$A_{7,2} = 25$$

$$B_{7,2} = 4336808689942017736029811203479766845703125$$

$$A_{7,3} = 25$$

$$B_{7,3} = 108420217248550443400745280086994171142578125.$$

When voter V_7 wants to vote in voting stage, he should login into system by his email and password. After that, the voter should determine his string of

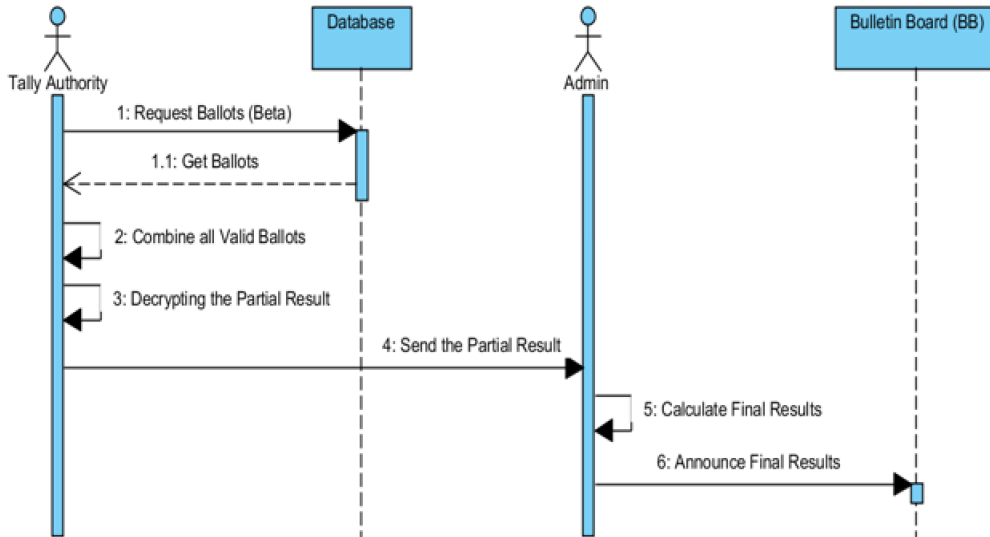


Figure 6: Tallying stage sequence diagram

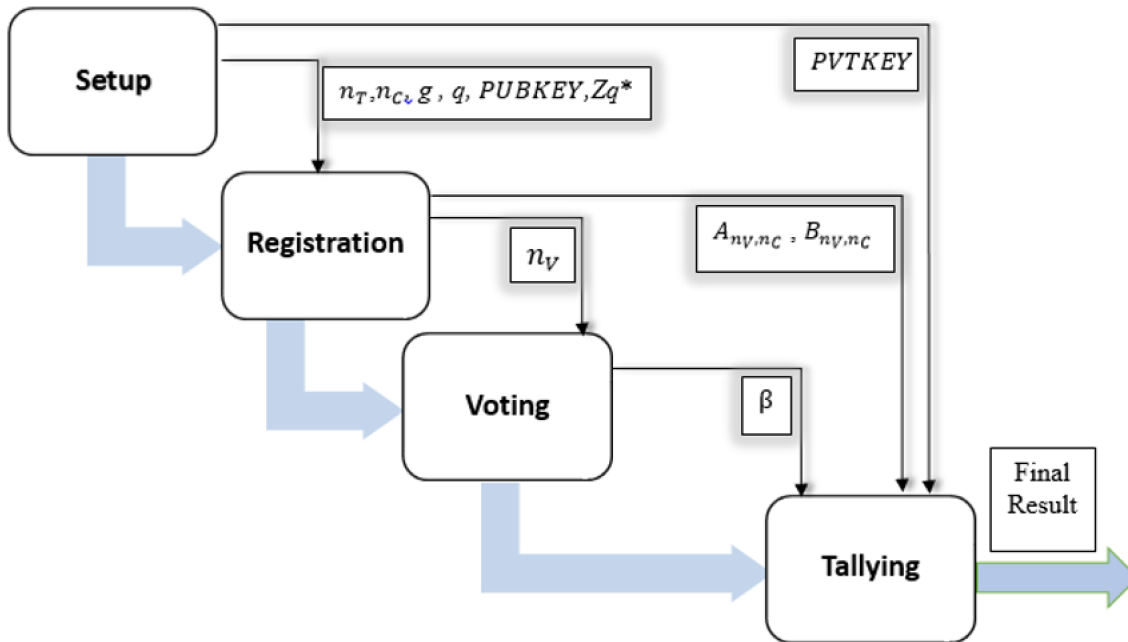


Figure 7: The structure of the e-Voting system


```

Output - Remote_e_Voting_Ihsan (run) X
Output - Remote_e_Vot

The Final Results :-
Candidate      Yes    No
=====
C[1]           3      7
C[2]           0     10
C[3]           7      3

BUILD SUCCESSFUL (total time: 0 seconds)

```

Figure 8: The final results of the example

- 1) **Eligibility:** Registrars Prove and confirm the eligibility of the person to vote. They denied from election any person who does not met the pre-defined requirements. In addition, there is a field called (isVoted) in class of voter. This field is marked when the voter casts his vote to prevent him from voting one more time.
- 2) **Privacy:** In the system, no one can link the identity of the voter and his vote, even the tally authorities. The privacy of voter is preserved by using homomorphic encryption protocol and reference technique.
- 3) **Accuracy:** The system can tally the valid votes with high accuracy by using homomorphic property of El-Gamal cryptosystem.
- 4) **Fairness:** No one can know the intermediate results or any partial result during election since the system is designed for multiple tally authorities. All tally authorities and admin should be jointly compute the final results.
- 5) **Receipt-freeness/coercion resistance:** The using of reference in the system make the system is receipt-freeness and that prevents the vote-buying. The voter cannot prove what is he voted in election to others.
- 6) **Mobility:** The system require the voter comes to specific locations only during registration days. The voter can cast his vote from anywhere he access to Internet during voting day.
- 7) **Simplicity/Convenience:** The system is relative simple. The user interface is user-friendly and not require high skills from the voters.
- 8) **Individual verifiability:** A voter must have the ability to verify that the vote he casted is accounted in the tally without any modification. The system achieved this property by publishing the β on (BB). Each voter can reach his β easily and check if it is changed or not.

- 9) **Scalability:** The practicality of a voting system depends on the factor of protocols complexity that used in the system. With aspect to storage, computation and communication, the system has to be scalable to any number of voters with more computations and hardware requirements.
- 10) **Availability:** In our system, the voters can be access all features during the election period.

7 Conclusions

In this paper, a practical secure e-Voting system is presented. By using homomorphic encryption, the system achieves the confidentiality of the voters. The system does not require a secure channel during a voting stage. The needless of encryption of votes allows the voter ensure that his vote is not changed. Unfortunately, e-Voting schemes based on homomorphic encryption require high computational space and time. The overhead for tallying is increasing depending on the increase in the number of voters, candidates and values of parameters. The system is practical with small and large scale elections with more computation as the election size increased. The system is implemented using JAVA, the language that deals with the huge numbers that consist of thousands of digits in both the integer and decimal form by using BigInteger and BigDecimal classes.

References

- [1] M. A. Based and S. F. Mjøl̄snes, "Security requirements for internet voting systems," in *Emerging Trends in Computing, Informatics, Systems Sciences, and Engineering*, pp. 519–530, Springer, 2013.
- [2] D. Boneh, E. Goh, and K. Nissim, "Evaluating 2-DNF formulas on ciphertexts," in *Theory of Cryptography Conference*, pp. 325–341, 2005.
- [3] Y. Chen, J. Jan, and C. Chen, "The design of a secure anonymous internet voting system," *Computers & Security*, vol. 23, no. 4, pp. 330–337, 2004.
- [4] B. Chevallier-Mames, P. Fouque, D. Pointcheval, J. Stern, and J. Traoré, "On some incompatible properties of voting schemes," in *Towards Trustworthy Elections*, pp. 191–199, 2010.
- [5] J. D. Cohen and M. J. Fischer, *A robust and verifiable cryptographically secure election scheme*, Department of Computer Science, Yale University, 1985.
- [6] J. C. Corena and J. A. Posada, "Multiplexing schemes for homomorphic cryptosystems," *Elements*, vol. 1, no. 1, 2013.
- [7] R. Cramer, R. Gennaro, and B. Schoenmakers, "A secure and optimally efficient multi-authority election scheme," *European Transactions on Telecommunications*, vol. 8, no. 5, pp. 481–490, 1997.

- [8] T. ElGamal, "A public key cryptosystem and a signature scheme based on discrete logarithms," in *Workshop on the Theory and Application of Cryptographic Techniques*, pp. 10–18, 1984.
- [9] C. Gentry, *A Fully Homomorphic Encryption Scheme*, PhD thesis, Stanford University, 2009.
- [10] V. George and M. Sebastian, "An adaptive indexed binary search tree for efficient homomorphic coercion resistant voting scheme," *International Journal of Managing Information Technology*, vol. 2, no. 1, pp. 1–9, 2010.
- [11] S. Goldwasser and S. Micali, "Probabilistic encryption & how to play mental poker keeping secret all partial information," in *Proceedings of the Fourteenth Annual ACM Symposium on Theory of Computing*, pp. 365–377, 1982.
- [12] H. Hussien and H. Aboelnaga, "Design of a secured e-voting system," in *International Conference on Computer Applications Technology (ICCAT'13)*, pp. 1–5, 2013.
- [13] A. Huszti, *A Homomorphic Encryption-based Secure Electronic Voting Scheme*, Faculty of Informatics, University of Debrecen, Hungary, 2011.
- [14] I. Jabbar and S. N. Alsaad, "Using fully homomorphic encryption to secure cloud computing," *Internet of Things and Cloud Computing*, vol. 4, no. 2, pp. 13–18, 2016.
- [15] M. F. Mursi, G. M. Assassa, A. Abdelhafez, and K. M. Abo Samra, "On the development of electronic voting: A survey," *International Journal of Computer Applications*, vol. 61, no. 16, 2013.
- [16] P. Paillier, "Public-key cryptosystems based on composite degree residuosity classes," in *International Conference on the Theory and Applications of Cryptographic Techniques*, pp. 223–238, 1999.
- [17] C. Porkodi, R. Arumuganathan, and K. Vidya, "Multi-authority electronic voting scheme based on elliptic curves," *International Journal of Network Security*, vol. 12, no. 2, pp. 84–91, 2011.
- [18] R. Rivest, S. Ledlie, et al., *Lecture Notes 15: Voting, Homomorphic Encryption*, 2002.
- [19] R. L. Rivest, L. Adleman, and M. L. Dertouzos, "On data banks and privacy homomorphisms," *Foundations of Secure Computation*, vol. 4, no. 11, pp. 169–180, 1978.
- [20] E. Saleh, "Processing over encrypted data: Between theory and practice," in *Proceedings of the 8th Ph.D. Retreat of the HPI Research School on Service-oriented Systems Engineering*, vol. 95, p. 163, 2015.
- [21] M. A. Will, B. Nicholson, M. Tiehuis, and R. K. Ko, "Secure voting in the cloud using homomorphic encryption and mobile agents," in *IEEE International Conference on Cloud Computing Research and Innovation (ICCCRI'15)*, pp. 173–184, 2015.
- [22] X. Yi and E. Okamoto, "Practical internet voting system," *Journal of Network and Computer Applications*, vol. 36, no. 1, pp. 378–387, 2013.
- [23] Y. Zhao, Y. Pan, S. Wang, and J. Zhang, "An anonymous voting system based on homomorphic encryption," in *10th International Conference on Communications (COMM'14)*, pp. 1–4, 2014.

Biography

Ihsan Jabbar received his bachelor degree in Computer Science from the College of Science/University of Al-Mustansiriyah in 2013. And his Master of Science degree in Computer Science from the same college.

Saad Najim Alsaad is a professor in Computer Science department at the College of Science/ University of Al-Mustansiriyah / Iraq. He is working as teaching staff member more than 20 years. He is interested in research domains such as: speech processing, information security, and object oriented software engineering. He is now working as Editor in Chief in Al-Mustansiriyah Journal of Science.

A Key-Policy Attribute-based Encryption Scheme for General Circuit from Bilinear Maps

Peng Hu and Haiying Gao

(Corresponding author: Haiying Gao)

Information and Technology Institute

No.62, Science Avenue, National High and New Technology

Industries Development Zone, Zhengzhou, Henan 450001, P. R. China

(Email: hupeng007@126.com)

(Received Oct. 6, 2016; revised and accepted Feb. 1 & Feb. 19, 2017)

Abstract

By access structure and attribute set, attribute-based encryption realizes fine-grained access control and one to many encryption. The expression of access structure directly decides the application range of one scheme and the general circuit reaches the best form. Since the safety of multilinear maps is suffered question, using relatively efficient and safe bilinear maps to construct circuit attribute-based encryption becomes popular. In this paper, we first propose a method that can convert any monotone circuit to an equivalent access tree. Then based on it, we propose a key-policy attribute-based encryption for general circuit from bilinear maps. Moreover, combining exiting method that can convert any access tree into LSSS structure and plenty of well-developed LSSS schemes, we can directly obtain corresponding circuit schemes. Compared with currently scheme from bilinear maps, our work is more efficient and expandable. In the standard model, selective security of our scheme is proved under the decisional bilinear Diffie-Hellman assumption.

Keywords: Attribute-based Encryption; Access Tree; Bilinear Maps; General Circuit; Selective Security

1 Introduction

With the Internet more and more developed, the application of cloud storing and cloud computing are more and more widely. And the security problem also becomes more and more serious [3, 21]. In traditional public key encryption, the message is encrypted to a specific individual. The efficiency becomes extremely low when sharing a message with multi users. Attribute-Based Encryption (ABE) as a new type public key primitive realizes one to many encryption and fine grand access control.

Sahai and Waters [18] proposed the concept of ABE in EUROCRYPT 2005. Different with traditional public key encryption, there are attribute set and access struc-

ture in encryption and key generation phase. And according to the position, the ABE can be divided into two types: when attribute set associate with ciphertext and access structure associate with private key, it is called Key-Policy ABE (KP-ABE); when attribute set associate with private key and access structure associate with ciphertext, it is called Ciphertext-Policy ABE (CP-ABE). Only when attribute set satisfy access structure, the user can decrypt the encrypted message. In 2006, Goyal et al. [10] proposed the first KP-ABE scheme, and the access structure of this scheme is access tree with highly efficient secret sharing approach.

How to improve the expression of the access structure is an important research field in ABE, and the progress of the improvement is really slow. After first access tree ABE was proposed in 2006, Lewko and Waters [12] converted it into Linear Secret Sharing Scheme (LSSS) as access structure until 2011. In 2013, Garg et al. [9] utilized the multilinear maps [8] built in ideal lattice to construct a KP-ABE scheme that supported general circuit as access structure. The general circuit can express any fixed running time program and reach the strongest expression in ABE [23]. After that, Tiplea et al. [20] proposed the first circuit KP-ABE scheme from bilinear maps. Until now, it is still the only one work that achieves the general circuit by bilinear maps. Recently, Hu and Jia [11] pointed out that the multilinear maps they used are not safe and gave a valid attack. Therefore, we build the scheme on mature bilinear maps and its assumption.

2 Related Work

Unlike other fast developing branches in the ABE system, the most important one which aim to achieve better expression improves really slowly. The first KP-ABE [10] scheme used access tree as structure in 2006. The access tree can be used to represent any monotone Boolean formulas which is a special case in mono-

tone circuit with limitation of fan-out one for every node. In 2011, [12] proposed a KP-ABE scheme with LSSS as structure, and in their scheme, they proposed a method that can convert any access tree into a LSSS matrix. Due the flexible and efficiency of LSSS, many schemes [1, 7, 13, 14, 15, 16, 17, 22] used LSSS structure to achieve additional property based on [12]. In 2013, [9] explained that the “backtracking attack” was the main barrier to extent access tree’s single fan-out into circuit’s multi fan-out. And they used level multilinear maps to prevent the attack, but the private key size and computation complexity of paring is extremely high. After that, there appeared many optimization and extensions schemes [4, 5, 6] based on [9]. However, [11] found a weakness on the multilinear maps they used and give a valid attack, leading their scheme unsafe. As for now, there is only one scheme achieved circuit ABE from bilinear maps. But its complexity is still high and has limitation on extension.

In this work, we propose a method that can converts any monotone circuit into a corresponding access tree, and then use the secret sharing method in [10] to construct a KP-ABE scheme. Selective security of our scheme in the standard model is proved under the decisional bilinear Diffie-Hellman assumption. Compared with [20], our scheme do not have extra gate (FANOUT gate) and its components, therefore our scheme is more efficient. More important, combining with the method proposed in [22] that converts any access tree into a corresponding LSSS, we can directly get plenty of circuit schemes with additional property based on current schemes like CP-ABE [24], private key tracing [14, 15], revoke [16], large universe [17], etc.

3 Preliminary

Definition 1. *Access Structure [19]: For a given non-empty finite set U , any non-empty subset S of U is called an access structure definite on U . S is called monotone if for $\forall B \in S$ satisfies:*

$$(\exists A \in S)(A \subseteq B) \Rightarrow B \in S.$$

If a subset of U also belongs to S , it is called authorized set; otherwise, it is called unauthorized set. In ABE scheme, we call the elements of U as attributes.

Definition 2. *Access Tree: An access tree is combined by leaf nodes and gates. Each leaf node has one outgoing wire and associates with one attribute. The gate type is either OR gate (1 of 2 threshold gate) or AND gate (2 of 2 threshold gate) which has two incoming wires and one out going wire.*

Definition 3. *General Circuit [9]: A general circuit is combined by input nodes and gates. Each input node has arbitrary numbers outgoing wires (at least one) and associates with one attribute. The gate type is either OR gate, AND gate which has two incoming wires, or NOT*

gate which has one incoming wire and one outgoing wire, and those gates can have arbitrary numbers outgoing wires (at least one).

In our scheme, we use notation $C_x(\Gamma_x)$ to represent a sub-circuit (sub-tree) with root node x , and abuse the subscript r to express entire circuit (tree). For input attribute set A , we use $C_x(A) = 1$ ($\Gamma_x(A) = 1$) to represent A satisfy sub-circuit (sub-tree) $C_x(\Gamma_x)$, and $C_x(A) = 0$ ($\Gamma_x(A) = 1$) to represent it is not. We use tuple (w, w_1, w_2) to represent a node w and its left and right child nodes w_1, w_2 , and use $S(w), R(w)$ to represent the sharing attaches and recovery value to outgoing wires of node w respectively.

Definition 4. *Monotone Circuit [9]: A general circuit is monotone if it does not have any NOT gate.*

Like [9] said, we can use De Morgan’s rule to convert any general circuit into an equivalent circuit with NOT gates only appear at input level, and above is a monotone one. Then we combine those NOT gates with attributes associated in input nodes. Therefore, we just consider monotone circuit in the scheme. In circuit or access tree, if the number of a node’s outgoing wire is one, we call it a single fan-out node; otherwise we call it a multi fan-out node.

3.1 Definition for Circuit KP-ABE

There are four algorithms in a KP-ABE scheme for circuit, including three probabilistic polynomial-time (PPT) algorithms and one deterministic polynomial-time (DPT) algorithm as follows:

Setup(λ, n) \rightarrow (PP, MSK): The setup is a PPT algorithm. It inputs the security parameter λ and number of system attribute n . It outputs the public parameters PP and master secret key MSK .

Encrypt(PP, A, m) \rightarrow CT : The encryption is a PPT algorithm. It inputs the public parameters PP , an attribute set A and a message m . It outputs ciphertext CT .

KeyGen(MSK, C) \rightarrow PK : The key generation is a PPT algorithm. It inputs the master secret key MSK and a circuit C . It outputs private key PK .

Decrypt(PK, E) \rightarrow m/\perp : The decryption is a DPT algorithm. It inputs private key PK and a ciphertext CT . It outputs a message m or the special symbol \perp .

3.2 Security Model for Circuit KP-ABE

The selective security model of circuit KP-ABE can be seen as a game between a challenger and an attacker. At the end, the attacker will give a guess. If the guess is right, the attacker wins the game; otherwise, the challenger wins.

Init. The attacker declares the attribute set A^* .

Setup. The challenger runs the Setup algorithm, then publishes the public parameters PP , and keeps the master secret key MSK .

Phase 1. The attacker requests any polynomial number times private key queries for any circuit C under the limitation that $C(A^*) = 0$, and the challenger return the corresponding private key to the attacker.

Challenge. The attacker issues two equal length messages m_0, m_1 , then challenger flips a random bit $b \in \{0, 1\}$, and return the corresponding ciphertext to the attacker.

Phase 2. Same as the **Phase 1**.

Guess. The attacker gives a guess b' of b . The advantage of the attacker in the game is defined by $\Pr[b = b'] - \frac{1}{2}$.

Definition 5. *Selective Security:* If for all PPT attackers at most have a negligible advantage in above game, we call this circuit KP-ABE scheme is selective secure.

3.3 Bilinear Maps and Assumption

Definition 6. *Bilinear Maps [2]:* For two multiplicative cyclic group of prime order p G_1, G_2 with generator g of G_1 and a map $e : G_1 \times G_1 \rightarrow G_2$. We call e is a bilinear map and G_1 is bilinear group if they satisfy:

- 1) *Bi-linearity:* for all $u, v \in G_1$ and $a, b \in \mathbb{Z}_p$, we have $e(u^a, v^b) = e(u, v)^{ab}$;
- 2) *Non-degeneracy:* $e(g, g) \neq 1$;
- 3) *Computable:* the group operation in G_1 and map e are both efficiently computable.

Definition 7. *Decisional Bilinear Diffie-Hellman (DBDH) Assumption:* Given two bilinear groups G_1, G_2 and $g, g^a, g^b, g^c, e(g, g)^{abc}, e(g, g)^z$ (a, b, c, z are randomly chosen from \mathbb{Z}_p), there is no polynomial-time algorithm can distinguish $e(g, g)^{abc}$ and a random element $e(g, g)^z$ in G_2 .

4 Circuit Conversion

The secret sharing of our scheme is based on [10]. But because of “backtracking attack”, it cannot directly used in circuit. Therefore, our idea is to convert each multi fan-out node into equivalent form of several single fan-out nodes, and then use the method in [10] to build a KP-ABE scheme while still resists attack. Our conversion’s direction is in a bottom up direction in circuit. For a node x with fan-out l ($l \geq 2$), we use l copies of sub-circuits C_x but only has single fan-out for root node, to replace the original sub-circuit C_x . Then move to the next node. For better understanding, we give a simple

example in Figure 1(abc). As shown in Figure 1(a), there are two multi fan-out nodes, and we first use two nodes 1 with single fan-out to link its two parents which turns into Figure 1(b). Then we use same method to convert AND gate in upper level and get circuit in Figure 1(c). We can easily find that Figure 1(a) and 1(c) are equivalent.

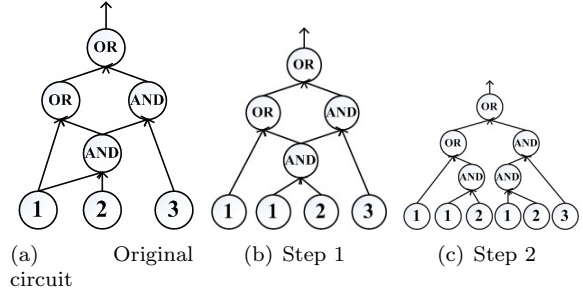


Figure 1: Circuit conversion

Here we explain why our scheme can resist the “backtracking attack”. The attack only takes place in multi fan-out gate that its outgoing wire links to an OR gate. As we can see in Figure 2, due to the sharing of incoming wires in OR gate and outgoing wires in multi fan-out gate (marked as X) are equal. When someone knows the left wire’s sharing of OR gate, it can directly know the left wire’s sharing of AND gate even though the multi fan-out gate is not satisfied and its sharing of outgoing wires is not supposed to know. In our scheme, we convert all multi fan-out gates to single fan-out. And in secret sharing phase, we attach different sharing to those wires; therefore the attacker cannot use the multi fan-out as bridge to attack other gates.

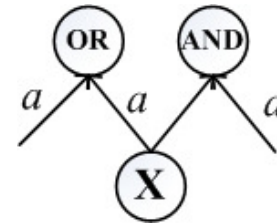


Figure 2: Backtracking attack

5 Our Construction

Setup(λ, n): In system setup phase, it inputs the security parameter λ to choose prime p and number of attributes n to choose attribute universe $U = \{1, \dots, n\}$. Then it generates two bilinear groups G_1, G_2 with order p and a bilinear map $e : G_1 \times G_1 \rightarrow G_2$. Suppose the generator of G_1 is g . At last, it randomly chooses $y \in \mathbb{Z}_p, t_i \in \mathbb{Z}_p$ for every $i \in U$ and publishes the public parameters:

$$PP = (Y = e(g, g)^y, (T_i = g^{t_i} | i \in U)) \quad (1)$$

keeps the master secret key:

$$MSK = (y, t_1, \dots, t_n) \quad (2)$$

Encrypt(m, A, PP): In encryption phase, it inputs the public parameters PP , an attribute set $A \subseteq U$ and a message $m \in G_2$. Then it randomly chooses $s \in Z_p$ and outputs the ciphertext:

$$\begin{aligned} CT &= (A, g^s, E', E_i) \\ E' &= me(g, g)^{ys} \\ E_i &= T_i^s = g^{st_i} | i \in A. \end{aligned}$$

KeyGen(C, MSK): In key generation phase, it inputs the master secret key MSK and a circuit C . Then it converts the circuit C into an equivalent access tree. After that, it sets $S(r) = y$ for the root node and shares the y in a top down manner as follows:

OR gate (w, w_1, w_2): If $S(w) = \delta$, then it sets:

$$S(w_1) = S(w_2) = \delta.$$

AND gate (w, w_1, w_2): If $S(w) = \delta$, then it randomly chooses $\varphi \in Z_p$ and sets:

$$\begin{aligned} S(w_1) &= \varphi, \\ S(w_2) &= \delta - \varphi. \end{aligned}$$

Finally, it generates the private key at each leaf node by the sharing it gets. For each leaf node x and its attached attribute t_x , it outputs the private key:

$$SK = \{SK_x = g^{S_x/t_x}\}.$$

Decryption(CT, PK): In decryption phase, it inputs the ciphertext CT with structure Γ and user's private key PK with attribute set A . Then it does the follow to calculate the message:

Leaf node (w): If $\Gamma_w(A) = 1$, it calculates:

$$\begin{aligned} R(w) &= e(SK_x, E_x) \\ &= e(g^{S_x/t_x}, g^{st_x}) \\ &= e(g, g)^{sS_x}. \end{aligned}$$

AND gate (w, w_1, w_2): If $\Gamma_w(A) = 1$, it calculates:

$$\begin{aligned} R(w) &= R(w_1) \cdot R(w_2) \\ &= e(g, g)^{\varphi s} e(g, g)^{\delta s - \varphi s} \\ &= e(g, g)^{\delta s}. \end{aligned}$$

OR gate (w, w_1, w_2): If $\Gamma_w(A) = \Gamma_{w_1}(A) = 1$, it sets:

$$R(w) = R(w_1).$$

Or if $\Gamma_w(A) = \Gamma_{w_2}(A) = 1$, it sets:

$$R(w) = R(w_2).$$

Finally, it will get $Y = e(g, g)^{ys}$ at the root node if $\Gamma(A) = 1$, and get message $m = E'/Y$.

6 Security Proof

In this section, we give the security proof of our KP-ABE scheme by DBDH assumption under the standard model. As described in Section 3.2, it is a game between a poly-time attacker and a challenger.

Theorem 1. *If there exists a poly-time attacker who can break our KP-ABE scheme with advantage ε , the challenger can solve the DBDH problem with advantage $\varepsilon/2$.*

Proof. The challenger first receives an instance of a BDHE assumption, which includes (g^a, g^b, g^c, T) and the challenger will decide whether $T = e(g, g)^{abc}$ or $T = e(g, g)^z$. Next it will use the attacker's ability to solve the problem.

Init. The attacker announces the challenge attribute set A^* .

Setup. The challenger sets $Y = e(g^a, g^b) = e(g, g)^{ab}$, then it randomly chooses r_i for all $i \in U$ and sets:

$$T_i = \begin{cases} g^{r_i}, & i \in A^* \\ g^{br_i}, & i \notin A^* \end{cases}$$

Finally it publishes the public parameters:

$$\begin{aligned} PP &= (p, G_1, G_2, g, e, n, Y, T_i) \\ Y &= e(g, g)^y \\ T_i &= g^{t_i} | i \in U. \end{aligned}$$

Phase 1. The attacker can submit any poly numbers circuits with limitation $C(A^*) = 0$. After receiving the circuit, the challenger converts it into access tree Γ and starts the secret sharing procedure.

The challenger first implicitly sets $y = S(r) = ab$ for the root node and sharing y by access tree in a top down manner as following (note that for a node w , if $\Gamma_w(A^*) = 0$, the sharing form of its outgoing wire would be an element in Z_p ; otherwise it would be an element in G_1).

OR gate (w, w_1, w_2): Suppose $S(w) = L$, it sets:

$$S(w_1) = S(w_2) = \delta.$$

AND gate (w, w_1, w_2): Suppose $S(w) = L$, it first randomly chooses $K \in Z_p$. Then if $\Gamma_w(A^*) = \Gamma_{w_1}(A^*) = \Gamma_{w_2}(A^*) = 1$, it sets:

$$\begin{aligned} S(w_1) &= K, \\ S(w_2) &= L - K. \end{aligned}$$

If $\Gamma_w(A^*) = 0, \Gamma_{w_1}(A^*) = 1, \Gamma_{w_2}(A^*) = 0$, it sets:

$$\begin{aligned} S(w_1) &= K, \\ S(w_2) &= L/g^K. \end{aligned}$$

If $\Gamma_w(A^*) = 0, \Gamma_{w_1}(A^*) = 0, \Gamma_{w_2}(A^*) = 1$, it sets:

$$\begin{aligned} S(w_1) &= L/g^K \\ S(w_2) &= K. \end{aligned}$$

If $\Gamma_w(A^*) = \Gamma_{w_1}(A^*) = \Gamma_{w_2}(A^*) = 0$, it sets:

$$\begin{aligned} S(w_1) &= g^K, \\ S(w_2) &= L/g^K. \end{aligned}$$

For each leaf node, it sets:

$$SK_x = \begin{cases} (g^b)^{S(x)/r_i}, & x \in A \\ S(x)^{1/r_i}, & x \notin A \end{cases}$$

At last, the challenger sends the private key $SK = \{SK_x\}$ to the attacker.

Challenge. The attacker submits two equal length messages m_0, m_1 to the challenger. Then the challenger flips a random coin $b \in \{0, 1\}$ and outputs the following ciphertext to the attacker:

$$E = (A^*, E' = m_v T, \{E_i = g^{cr_i}\}_{i \in A^*}).$$

Phase 2. This phase is same as **Phase 1**.

Guess. The attacker gives a guess b' about b . If $b' = b$, the challenger decides $T = e(g, g)^{abc}$; otherwise, it decides $T = e(g, g)^z$.

Next, we calculate the advantage that challenger has. We use $\Pr[C]$ to represent the probability that the challenger's decision is right, use $\Pr[C_{abc}]$ to represent the probability that the challenger decide $T = e(g, g)^{abc}$ and use $\Pr[C_z]$ to represent the probability that challenger decides $T = e(g, g)^z$. Suppose the attacker can break this scheme with advantage ε , then:

$$\begin{aligned} \Pr[C] &= \Pr[C_{abc}|T = e(g, g)^{abc}] \\ &\quad \cdot \Pr[T = e(g, g)^{abc}] \\ &\quad + \Pr[C_z|T = e(g, g)^z] \cdot \Pr[T = e(g, g)^z] \\ &= \Pr[b' = b|T = e(g, g)^{abc}] \\ &\quad \cdot \Pr[T = e(g, g)^{abc}] \\ &\quad + \Pr[b \neq b|T = e(g, g)^z] \cdot \Pr[T = e(g, g)^z] \\ &= \frac{1}{2} \left(\frac{1}{2} + \varepsilon \right) + \frac{1}{2} \times \frac{1}{2} \\ &= \frac{1}{2} + \frac{\varepsilon}{2}. \end{aligned}$$

7 Efficiency Analysis

In this section, we give the efficiency analysis by comparing our scheme with [20] since it is the only one work that

achieved circuit ABE from bilinear maps. The efficiency of [20] and our scheme both rely on the distribution and numbers of multi fan-out nodes in circuit. Therefore, we give the comparison in a more concrete circuit as follows.

Table 1: Private key size in [20] and our scheme

Scheme	Worst case	Best case
[20]	$nj + n + j^r$	$nj + n + r(j - 1)$
Our	$n + j^r$	$n + r(j - 1)$

Suppose there are n input nodes, r multi fan-out nodes all with j outgoing wires. The best case in both [20] and our is that there is no path between any two multi fan-out nodes, and the private key size of [20] is $nj + n + r(j - 1)$ and our is $n + r(j - 1)$. The worst case is that there is a path through all multi fan-out nodes, and the private key size of [20] is $nj + n + j^r$ and our is $n + j^r$. The private key size is between this two in other cases. We give a summary in the Table 1. The private key size also means the paring times in decryption phase, therefore our scheme is more efficiency than [20].

8 Conclusion

In this work, we first propose a method that can convert any monotone circuit into an equivalence access tree, and then based on that, we propose a KP-ABE scheme for general circuit from bilinear maps that can resist "backtracking attack", and prove its selective security under DBDH assumption in standard model. Compared with the only one circuit KP-ABE from bilinear, our scheme is more efficient than that. More important, based on existing method that can convert any access tree into LSSS matrix and plenty of efficient LSSS ABE schemes with different additional property, we can directly obtain the corresponding circuit ABE schemes.

Currently, multilinear maps are not safe and the complexity of circuit ABE from bilinear maps are still too high for practical use. How to optimize the secret sharing procedure for circuit still need further research.

Acknowledgments

The authors would like to thank the anonymous reviewers of this paper for their valuable comments and suggestions. This work was sponsored in part by the National Natural Science Foundation of China [Grant No.61272041, Grant No.61601515], Foundation of Science and Technology on Information Assurance Laboratory [Grant No.KJ-15-006] and Fundamental and Frontier Technology Research of Hennan Province (Grant No.162300410192).

References

- [1] 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.
- [2] D. Boneh and M. K. Franklin, "Identity-based encryption from the weil pairing," in *Advances in Cryptology (CRYPTO'01)*, pp. 213–229, California, USA, Aug. 2001.
- [3] Z. Cao, C. Mao, 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.
- [4] P. Datta, R. Dutta, and S. Mukhopadhyay, "Compact attribute-based encryption and signcryption for general circuits from multilinear maps," in *Progress in Cryptology (INDOCRYPT'15)*, pp. 3–24, India, Dec. 2015.
- [5] P. Datta, R. Dutta, and S. Mukhopadhyay, "General circuit realizing compact revocable attribute-based encryption from multilinear maps," in *18th International Conference on Information Security (ISC'15)*, pp. 336–354, Norway, Sept. 2015.
- [6] C. C. Dragan and F. L. Tiplea, "Key-policy attribute-based encryption for general boolean circuits from secret sharing and multi-linear maps," in *Second International Conference on Cryptography and Information Security in the Balkans*, pp. 112–133, Slovenia, Sept. 2015.
- [7] X. B. Fu, S. K. Zeng, and F. G. Li, "Blind expressive ciphertext policy attribute based encryption for fine grained access control on the encrypted data," *International Journal of Network Security*, vol. 17, no. 6, pp. 661–671, 2015.
- [8] S. Garg, C. Gentry, and S. Halevi, "Candidate multilinear maps from ideal lattices," in *Advances in Cryptology (EUROCRYPT'13)*, pp. 1–17, Greece, May 2013.
- [9] S. Garg, C. Gentry, S. Halevi, A. Sahai, and B. Waters, "Attribute-based encryption for circuits from multilinear maps," in *Advances in Cryptology (CRYPTO'13)*, pp. 479–499, CA, USA, Aug. 2013.
- [10] V. Goyal, O. Pandey, A. Sahai, and B. Waters, "Attribute-based encryption for fine-grained access control of encrypted data," in *Proceedings of the 13th ACM Conference on Computer and Communications Security (CCS'06)*, pp. 89–98, Alexandria, VA, USA, 2006.
- [11] Y. P. Hu and H. W. Jia, "Cryptanalysis of GGH map," in *Advances in Cryptology (EUROCRYPT'16)*, pp. 537–565, Vienna, Austria, May 2016.
- [12] A. B. Lewko and B. Waters, "Decentralizing attribute-based encryption," in *Advances in Cryptology (EUROCRYPT'11)*, pp. 568–588, Tallinn, Estonia, May 2011.
- [13] Q. Y. Li and F. L. Zhang, "A fully secure attribute based broadcast encryption scheme," *International Journal of Network Security*, vol. 17, no. 3, pp. 255–263, 2015.
- [14] Z. Liu, Z. F. Cao, and D. S. Wong, "Fully collusion-resistant traceable key-policy attribute-based encryption with sub-linear size ciphertexts," in *10th International Conference on Information Security and Cryptology*, pp. 403–423, Beijing, China, Dec. 2014.
- [15] J. T. Ning, X. L. Dong, Z. F. Cao, L. F. Wei, and X. D. Lin, "White-box traceable ciphertext-policy attribute-based encryption supporting flexible attributes," *IEEE Transactions on Information Forensics and Security*, vol. 10, no. 6, pp. 1274–1288, 2015.
- [16] T. Okamoto and K. Takashima, "Fully secure unbounded inner-product and attribute-based encryption," in *Advances in Cryptology (ASIACRYPT'12)*, pp. 349–366, Beijing, China, Dec. 2012.
- [17] Y. Rouselakis and B. Waters, "Practical constructions and new proof methods for large universe attribute-based encryption," in *ACM SIGSAC Conference on Computer and Communications Security (CCS'13)*, pp. 463–474, Berlin, Germany, Nov. 2013.
- [18] A. Sahai and B. Waters, "Fuzzy identity-based encryption," in *Advances in Cryptology (EUROCRYPT'05)*, pp. 457–473, Aarhus, Denmark, May 2005.
- [19] D. R. Stinson, *Cryptography: Theory and Practice. (3ed, in English)*, Britain: Chapman and Hall, 2005.
- [20] F. L. Tiplea and C. C. Dragan, "Key-policy attribute-based encryption for boolean circuits from bilinear maps," in *First International Conference on Cryptography and Information Security in the Balkans*, pp. 175–193, Istanbul, Turkey, Oct. 2014.
- [21] Z. Wang, Y. Lu, G. Sun, "A policy-based deduplication mechanism for securing cloud storage," *International Journal of Electronics and Information Engineering*, vol. 2, no. 2, pp. 70–79, 2015.
- [22] B. Waters, "Ciphertext-policy attribute-based encryption: An expressive, efficient, and provably secure realization," in *14th International Conference on Practice and Theory in Public Key Cryptography (PKC'11)*, pp. 53–70, Taormina, Italy, Mar. 2011.
- [23] B. Waters, "Functional encryption: Origins and recent developments," in *16th International Conference on Practice and Theory in Public-Key Cryptography (PKC'13)*, pp. 51–54, Nara, Japan, Feb. 2013.
- [24] J. Xu, Q. Y. Wen, W. M. Li, and Z. P. Jin, "Circuit ciphertext-policy attribute-based hybrid encryption with verifiable delegation in cloud computing," *IEEE Transactions on Parallel Distributed Systems*, vol. 27, no. 1, pp. 119–129, 2016.

Biography

Peng Hu received his BS degree in information security from Zhengzhou Information and Technology Institute, Hennan, China, in 2014. He is currently a postgraduate student in Zhengzhou Information and Technology Institute. His current research interests include cryptography and information security.

Haiying Gao received her BS degree in mathematics from Hennan university, Hennan, China, in 1999. She received her MS degree in information security from Zhengzhou Information and Technology Institute, Hennan, China, in 2003, and her PhD from Xidian University, Shanxi, China, in 2006. She is currently a professor with the Zhengzhou Information and Technology Institute, Hennan, China. Her research interest focuses on cryptology theory.

An Anti-Phishing Password Authentication Protocol

Pramote Kuacharoen

Department of Computer Science, Graduate School of Applied Statistics

National Institute of Development Administration

118 SeriThai Rd., Bangkapi, Bangkok 10240, Thailand

(Email: pramote@as.nida.ac.th)

(Received Aug. 31, 2016; revised and accepted Jan. 15 & Feb. 20, 2017)

Abstract

Password authentication is commonly used to authenticate the user in web-based services such as internet banking due to its simplicity and convenience. Many users have multiple accounts and use the same password. The password is usually sent to the server over an HTTPS connection. However, this common practice makes the system vulnerable. An attacker can set up a phishing site masquerading as the genuine site and attempts to steal the user's credentials. If the user's credentials are successfully stolen, all accounts are compromised. Moreover, since passwords are common, a break-in to a system that is not well protected might cause a cascaded break-in. This paper describes an authentication protocol which enables the user to securely use the same password for multiple servers, and protects against phishing attacks. The protocol also allows multiple authentication sessions simultaneously while preventing replay attacks. Furthermore, the protocol is also resilient against denial-of-service attacks since no state is maintained on the server during the authentication process.

Keywords: Anti-phishing; Authentication; Mutual Authentication; Password

1 Introduction

Phishing is a technique that employs both social engineering and technical subterfuge to steal personal identifiable information and financial account credentials. The criminal creates a replica of an existing web page to deceive the victims [3]. Usually, the criminal sends emails which resemble emails from legitimate entities to potential victims. Unaware of criminal activities, the victims click the link in the email to visit the website where they are asked to provide personal information such as username, password, and credit card number. The criminal records this information and uses it to impersonate the victims or to commit financial fraud [15, 16].

Although the phishing site appears to be similar to the

legitimate site, the Uniform Resource Locator (URL) is different, usually suspicious. The phishing site is short lived so that it cannot be effectively blacklisted. An experimental phishing attack was performed at Indiana University targeting students aged 18 to 24 years old [9]. The acquaintance data are harvested from social network websites. The experiment spoofed an email message between two friends. Experiments showed that 72% of students entered their secure university credentials into the spoofed site whose domain name was clearly distinct from Indiana University.

When the user moves the pointer to hover over a hyperlink, the URL is usually shown on the status bar. A user with this knowledge makes an attempt to verify the destination URL using this method as a safeguard against phishing. However, a status bar can be easily spoofed. The criminal can use a simple *onclick* event to change the destination URL.

Many web browsers have anti-phishing features built in. However, some users fail to notice the warning, do not understand the warning, or ignore the warning [5]. In order to provide the anti-phishing features, the web browsers must maintain the list of the phishing sites. As aforementioned, phishing sites cannot be effectively blacklisted and the user is not protected until the phishing site is included on the list.

Several large financial institutions, including Bank of America and The Vanguard Group, attempt to combat against phishing attacks by implementing a technique called SiteKey which is the product of RSA Data Security. SiteKey uses the following challenge-response technique:

- 1) The customer identifies himself by submitting the username. If the username is valid, the site continues to the next step. Otherwise, the site displays an error message indicating that the username is not correct.
- 2) The site authenticates itself to the customer by displaying an image and a phrase that the user has previously chosen. If the user does not recognize them, the user should assume that the site is a phishing

site and should not proceed. If the user recognizes the displayed information, the user may consider that the site is authentic.

- 3) The user authenticates oneself by supplying the password. If the password is correct, the user is authenticated.

In practice, SiteKey is ineffective [19, 24]. People do not notice or do not care when the SiteKey is missing. Moreover, SiteKey technique has a security design flaw. The criminal can learn whether or not the username exists. The rationale of SiteKey is that the phishing site does not have the customer's SiteKey. However, the phishing site can obtain the correct SiteKey from the genuine site, and then displays it to the user.

The Anti-Phishing Working Group (APWG) reported that phishing attack numbers declined 20 percent from late 2012 to early 2013. This was due to a precipitous drop in virtual server phishing attacks, where the criminal seizes control of a web server that hosts many unique domains and then creates phishing pages for those domains [7]. According to APWG, trends indicate phishing levels returning to the levels seen prior to the record-setting highs of 2015. Therefore, these criminal activities are still prevalent and an effective anti-phishing attack technique is needed.

The purpose of this research is to design and implement an authentication protocol which is secure and protects the user against phishing attacks. The following requirements are the design goals of the anti-phishing password authentication protocol.

- The protocol must protect users against phishing attacks.
- The protocol must allow users to safely use the same password across many websites.
- The protocol must achieve user authentication without reviewing the password to the server at any point.
- The protocol must be secure against known attacks.

This paper consists of five sections. Section 1 introduces the motivation of the paper. Section 2 describes background information and related work in the area of phishing and password authentication. Section 3 presents the design of the anti-phishing password authentication protocol. Section 4 provides the security analysis of the protocol. Finally, Section 5 concludes the paper.

2 Background and Related Work

This section provides background and related work which includes phishing, password authentication, cryptographic challenge-response authentication, and existing anti-phishing password-based protocols.

2.1 Phishing

Phishing is the attempt to obtain sensitive information such as usernames, passwords, and financial information by masquerading as a trustworthy entity in electronic communication [14]. Phishers use social engineering schemes using spoofed emails purporting to be from legitimate businesses and agencies. The schemes are designed to lead victims to counterfeit websites and deceive the victims into divulging sensitive information.

APWG publishes quarterly phishing attack trends reports. Figure 1 shows the phishing trends. The total number of unique phishing reports received has sharply risen from year 2012 to year 2015. Phishing has been increasingly threatening individuals.

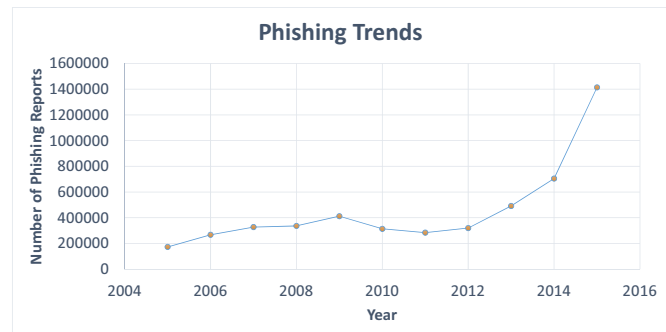


Figure 1: Phishing trends

Phishers usually send emails impersonating trusted entities luring victims to visit phishing sites. Phishing targets the user who has no knowledge about social engineering attacks or internet security [8]. Figure 2 shows an example phishing email which pretends to be from PNC Bank asking its customer to sign in by clicking on the link. The link displays the URL of PNC Bank. However, when the victim clicks on the link, the phishing site is shown. An unsuspecting customer who has an account with PNC Bank would sign in as instructed in the email. By doing so, the customer unknowingly gives the phisher their account credentials.

For an HTML page, the displayed link and the actual link can be different. When user moves a pointer over the link, the status bar shows the actual URL. This may give some confidence to the user who is familiar with browsing the Web. However, a status bar can be programmed to display whatever the phisher desires. A user with some technical knowledge is likely to be a victim for a status bar spoofing. When the user clicks the link, the actual link shows on the browser's address bar. The phisher tricks the victim by using a site name similar to the real site by misspelling the name. For example, the phisher may use letter 'a' instead of letter 'o', 'l' instead of letter '1', or '0' instead of letter 'o'. When the user glances at the address bar, the user assumes the website is legitimate. The phisher may also employ a poorly written redirection program from the real website to the phishing site. The

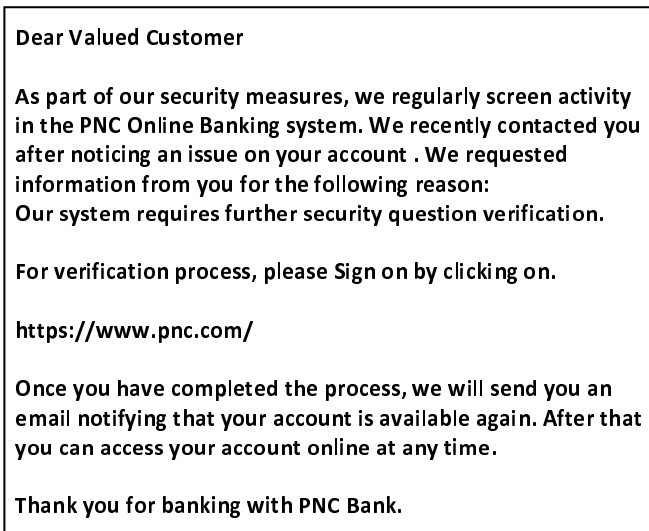


Figure 2: An example of phishing email

address bar will display the real website address briefly and then it will display the phishing site address. This will make the victim think that displayed website is redirected from the trusted site and trust the displayed site.

Online security warnings have historically failed because users do not understand or believe them. Since phishing is prevalent, new online security warnings have been redesigned. It is difficult to automatically detect phishing with accuracy. Therefore, anti-phishing tools use warnings to alert the users to potential phishing sites, rather than blocking them. There are two types of warnings, namely, passive warning and active warning. A passive warning indicates a potential danger by changing colors, providing textual information, or by other means without interrupting the user's task. However, research has shown that passive warnings are failing users because users often fail to notice them or do not trust them. An active warning, on the other hand, forces users to notice the warning by interrupting them. Many popular web browsers include active phishing warnings since research has shown that passive warnings are often ignored.

Despite growing efforts to educate users and to create better detection tools, users are still susceptible to phishing attacks. Phishing can deceive users because the users are willing to trust websites that appear to be designed well and look familiar.

Many research papers have proposed methods of detecting phishing sites based on URLs of web pages [1, 4, 10, 23]. After confirming that the website is probably a phishing site, a security warning is issued to the user. However, using phishing site detection methods does not guarantee that the algorithms are always accurate.

2.2 Password Authentication

Password authentication is the simplest form of an authentication model. The user presents the username and

the password to the authenticating entity [13, 20]. The password authentication is commonly used in authenticating the user over the Internet. The server needs to store the user's password in order to authenticate the user [17]. The password must be protected. Storing the password in clear text is inadvisable. This is because a compromised user database file reveals all passwords. The attacker may be able to obtain the password database through an SQL injection attack [2, 12]. Instead, the hash value of the password should be saved. A weakness of using the hash value is that two users with the same password have an identical hash value. Furthermore, the attacker can use a dictionary attack against the entire user database. The best way to protect a password is to employ salted password hashing. The server randomly generates a number called salt and calculates the hash of the salt and the password. Therefore, two users with the same password have different salted passwords. The server stores the salt and the salted password along with other information. Upon logging in, the user supplies the username and the password. The server computes the hash value of the salt and the received password, and then compares the resulting value to the stored hash value. If the two values are identical, the user is authenticated.

Sending a password in clear text is vulnerable to eavesdropping. Using an SSL connection helps protect the conversation during transit. However, it does not prevent phishing attacks.

2.3 Cryptographic Challenge-Response Authentication

In this type of authentication, the user and the system share some secret [21]. For two-way authentication, both the user and the system must convince each other that they know the shared secret without transmitting the secret in the clear text over the communication channel. To accomplish this, the server encrypts the randomly generated number or nonce and sends it to the user as a challenge. The user must return a corresponding response which is calculated from decrypting the challenge and encrypting the value derived from the decrypted value. This proves that the user has the ability to decrypt the challenge. Therefore, the user knows the shared secret.

A drawback of the challenge-response authentication is that it can be defeated by man-in-the-middle attacks. For example, the user visits a phishing site and submits his username. The phishing site forwards the user's identity to the genuine site impersonating the user. The server sends a challenge in order to authenticate the user. The attacker presents the challenge to the user to obtain the correct response which is subsequently sent to the server.

2.4 Existing Anti-phishing Password-based Protocols

Rose et al. present a method to improve password security and to defend against password phishing [18]. The server

stores the hash of the user's password and the domain name. When the user enters the username and the password is prefixed with two escape characters, the browser extension applies a Pseudo Random Function (PRF) to a combination of the password and the domain name. The username and the hash value are sent to the server. The domain name is automatically obtained. If the user enters the credentials on the phishing page, the phisher cannot obtain the clear text password. Moreover, the hash value is different from the one stored on the actual server since the phishing site has a different domain name. Using domain name as salt has a drawback. The attacker may compromise a server under the same domain and may set up a phishing page. The correct salted password can be captured. Since password and domain name remains unchanged, the salted password is the same, making it susceptible to replay attacks.

In [6], a protocol that allows a client to securely use a single password across multiple servers and prevents phishing attacks is proposed. The client can be authenticated without revealing the password to the server at any point. The protocol employs a one-time ticket technique. The client sets the next authentication ticket. The ticket consists of the hash of the random number, the password, and the server name. The client identifies himself by sending the identity. The server challenges with the previously stored random number. Subsequently, the client computes the ticket using the received number. The client also randomly generates a number and uses it to create the next authentication ticket. The client responds with the current ticket, the next challenge random number, and the hash value of the next authentication ticket. Although the clear text password remains unchanged, the ticket changes each time the client is authenticated. This is equivalent to changing the password at the server every time the user signs in, which makes the protocol susceptible to message modification attacks. Consider the scenario where an attacker intercepts the response from the client. The attacker then can create a ticket using his password and replaces the hash value with the one generated from his ticket. The server has no way to verify the authenticity of the hash value. Hence, the attacker can log in.

3 Design of the Protocol

This section describes the design of the anti-phishing password authentication protocol. The design objectives and the notions are explained. Then, this section discusses the password storage and the authentication that are designed to meet the objectives.

3.1 Design Objectives

The primary objective of this paper is to design an authentication protocol which is secure and protects the user against phishing attacks and other known attacks. The

following requirements are the design goals of the anti-phishing password authentication protocol.

- 1) The protocol must protect users against phishing attacks.
- 2) The protocol must allow users to safely use the same password across many websites.
- 3) The protocol must achieve user authentication without revealing the password to the server at any point.
- 4) The protocol must be secure against known attacks such as password database attacks, server spoofing attacks, and denied of service attacks.
- 5) The protocol must allow multiple authentication sessions.
- 6) The protocol must be mutual authentication.

3.2 Notations

Table 1 shows notations which are used throughout this paper.

Table 1: Notations and description

Notations	Descriptions
C	Client
S	Server
AD_S	Server's address, i.e., IP address
AD_C	Client's address, i.e., IP address
N_1, N_2, N_3	Nonce
$HMAC(K, M)$	Keyed-hash message authentication code function, where K is the secret key and M is the message
$Times$	Session valid time which consists of the start time and the expiration time
SAC	Session authentication code
\parallel	Concatenation operator
\oplus	XOR operator

3.3 Password Storage

It is crucial that the user's credentials are protected even though the user database has been compromised. The attacker should not gain knowledge from it. Therefore, the password should not be stored or sent as clear text. Traditionally, for each user, the server stores the username, the salt, and the hash value of the salt and the user's password. This protects the user's credentials and defends against dictionary attacks and pre-computed rainbow table attacks. However, both username and password are sent to the server to be authenticated. For a valid username, the server calculates the hash value of the salt and the received password and compares the resulting value

with the one stored on the server. Since the actual password is transmitted to the server, the attacker can obtain this information from a compromised server and can use the user's credentials to gain access to other servers. Therefore, the user's password should not be transmitted as clear text.

If the password is hashed at the client machine and sent to the server, the client-side hash logically becomes the user's password. Therefore, it is equivalent to storing passwords in clear text. If the attacker obtained the hash value, the attacker can use it to authenticate to the server. Hence, the server must store a value which is derived from the received value.

To achieve the design objectives, the authenticator must be derived from the user's password and must be server specific. Table 2 shows the user database. Each row consists of username, salt, and masked secret. The salt is credential specific. In other words, each user is randomly assigned a salt. This prevents a dictionary attack on the entire database. The attacker must pick an individual to attack. The masked secret is calculated by XORing the user's secret and the mask. The user's secret is derived from the username, the password, and the Fully Qualified Domain Name (FQDN), i.e., Hash(username || password || FQDN). The mask is the hash value of the server's secret and the user's salt, $HMAC(K_S || salt)$. The server's secret is not stored on disk. It is inputted when the server starts. Therefore, a compromised database does not reveal the server's secret or the user's secret.

HMAC is a message authentication code based on a cryptographic hash function [11]. The length of the authentication code is fixed. Only the parties, who have the knowledge of the secret key, can produce a valid message authentication code. The advantage of using HMAC is that the cryptographic hash function generally executes more quickly in software than symmetric and asymmetric ciphers [22].

Table 2: User database

Username	Salt	Masked Secret
u_1	$salt_1$	$K_{c1} \oplus HMAC(K_s, salt_1)$
u_2	$salt_2$	$K_{c2} \oplus HMAC(K_s, salt_2)$
u_3	$salt_3$	$K_{c3} \oplus HMAC(K_s, salt_3)$
...

3.4 The Protocol

In an insecure network environment, any client can connect to a server. The obvious risk is user impersonation. An attacker can pretend to be another user and obtain unauthorized access. To counter this threat, the server must be able to authenticate the user requesting the service.

Figure 3 summarizes the basic authentication dialog. The following is the brief description of the protocol.

- 1) The client identifies itself to the server by sending the user's ID, the address of the server, and a random value N_1 .
- 2) The server replies back with received information, another random value N_2 , and the start time and the expiration time of the authentication session. Moreover, the server includes the server's authenticator and the Session Authentication Code (SAC).
- 3) The client responds with the user's ID, server address, N_2 , another random value N_3 , *Times*, the client's authenticator, and the session authentication code.

1. C → S: ID_c || AD_s || N₁
2. S → C: ID_c || AD_s || N₁ || N₂ || Times || Authenticator_s || SAC
3. C → S: ID_c || AD_s || N₂ || N₃ || Times || Authenticator_c || SAC
Authenticator_s = HMAC(K_s, ID_c || AD_s || N₁ || N₂ || Times)
Authenticator_c = HMAC(K_c, ID_c || AD_s || N₂ || N₃ || Times)
SAC = HMAC(K_s, ID_c || AD_s || N₂ || Times)

Figure 3: Summary of the authentication exchanges

When the user connects to a server to use a service, the client software C in the user's computer requests the username and password, and then sends a message to the server S that includes the user's ID, the server's address, and the random nonce N_1 . The server first verifies if the requested server's address belongs to the server, and checks its database to see if the user exists. If the user is in the database, the server obtains the salt and the masked secret of the user. The mask is generated from hashing the server's secret and the user's salt. The resulting value is then XORed with masked secret to obtain the user's secret. Next, the server randomly generates another nonce N_2 and sets the start time and the expiration time for the authentication session.

The server constructs the response message which includes user's ID, the server's address, the received nonce N_1 , the generated nonce N_2 , and the times. The server authentication which is used to authenticate itself to the client is computed using the HMAC algorithm with the response message and the user's secret. The session authentication code for the session is also generated using the HMAC algorithm with the server's secret and the message which includes the user's ID, the client address, the nonce N_2 , and the times. The server sends the response message, the server authentication, and the session authentication code. After replying, the server can discard the calculated values.

When the client receives the response, the client verifies that the response is corresponding to the request by checking if the response message contains the requested

information. The server authenticator is then verified. A valid server authenticator proves that the server knows the user's secret. Now that the client has the challenge information, authenticating itself can be done next. The client creates a response message which consists of the user's ID, the server address, the received nonce N_2 , the nonce N_3 , and the times. The client also creates the client authentication by using the HMAC algorithm with the response message and the user's secret key. Subsequently, the client sends the response message, the client authenticator, and the session authentication code to the server.

Upon receiving the response, the server checks to ensure that the response is intended for the server and has not expired. If the user's ID is in the database, the server derives the user's secret from the masked secret, the user's salt, and the server's secret as previously described. Afterward, the server verifies the client authenticator. Successful verification implies that the user knows the password which is a crucial component in creating the client secret. The server then validates the session authentication code. A valid SAC is the SAC which has not expired and has not been used. The server saves the SAC which has been used to the user's SAC list. Future sessions will be checked against this list to ensure that the SACs are used only once. To perform the validation task efficiently, expired SACs are removed from the SAC list.

Table 3 summarizes the justification for each of the elements in the protocol.

4 Security Analysis

This section analyzes the security of the proposed protocol which includes the security of passwords, security of the server's secret key, and security of the communication protocol.

4.1 Security of Passwords

The obvious approach to password attack is to guess the password. The two most common methods of guessing passwords are brute-force attacks and dictionary attacks. These two types of attacks can be performed online or offline as described in the following section.

4.1.1 Online Brute-Force and Dictionary Attacks

In the brute-force approach, the attacker tries all possible passwords. On average, an attacker will have to try half of all possible combinations before finding the correct password. To defend against such an attack, the password length policy must be enforced. The password must be at least eight characters long. A longer password is a better password. Moreover, for online password guessing, the system should be configured to slow the attack by delaying between successive login attempts and limiting the number of unsuccessful attempts before disabling the account for a period of time or indefinitely until the account is reset.

The attacker impersonates the user and attempts to log in on a server by trying all passwords in an exhaustive list called a dictionary. An online dictionary attack feeds a server with thousands of username and password combinations. To protect against dictionary attacks, the user must use a strong password which can be enforced by access policy. Guidelines that are designed to make passwords less easily discovered by intelligent guessing and cracking tools include using complex passwords with an appropriate length. A complex password is a password which uses several types of keyboard characters. As a result, complex passwords are unlikely to be in the attacker's dictionary. The number of unsuccessful login attempts should also be limited as aforementioned.

4.1.2 Offline Brute-Force and Dictionary Attacks

The attacker obtains the user database and attempts to perform offline brute-force attacks or dictionary attacks. Each entry in the user database consists of the username, the salt, and the masked secret. The masked secret is calculated from the user's secret and the mask. However, the mask is user specific and depends on the server's secret. Therefore, this makes it impossible to use lookup tables and rainbow tables to crack the password.

4.2 Security of the Server's Secret Key

The server's secret key is used to create the user mask and the session authentication code using the HMAC algorithm. The user mask is not stored in the user database. It is XORed with the user's secret. If the attacker obtains the user database, the mask is not readily available. However, the attacker can register an account. Since the attacker can compute his own secret key, the mask can be obtained. Breaking the server's secret key would only compromise all users' secrets on a specific server. However, if the attacker breaks the server's secret key without having the user database, the attacker will not be able to impersonate other users. The security of the server's secret key depends entirely on the security of the HMAC algorithm.

Attacks on HMAC can be grouped in two categories, namely; brute-force attacks and cryptanalysis. The level of effort for brute-force attack on the HMAC algorithm has a similar level to that for symmetric encryption algorithms. Cryptanalysis attacks on the HMAC algorithm. The security of the HMAC depends in some way on the cryptographic strength of the underlying hash function. HMAC is considered secure. Therefore, it is computationally infeasible for the attacker to derive the server's secret key.

4.3 Security of the Communication Protocol

The authentication exchanges are done over the HTTPS protocol. However, the proposed authentication protocol

Table 3: Rationale for the elements of the protocol

Message 1	Client requests an authentication session
ID_C	Tells the server's identity of the user from this client
AD_S	The perceived server's address by the client, i.e., the IP address
N_1	A random value to be repeated in message 2 to assure that the response is fresh and has not been replayed by the attacker
Message 2	Server returns a session and authenticates itself to the client
ID_C	Indicates the rightful owner of the session
AD_S	The server's address
N_1	Nonce from message 1
N_2	A random value to ensure that the response is fresh. It is also used to generate the ticket authentication code
$Times$	Provides time sensitive authentication
$Authenticator_S$	Proves that the server knows the client's secret and the information has not been modified
SAC	The session authentication code to be repeated in message 3 to ensure that the session is authentic and is used only once within the time limit
Message 3	Client authenticates itself to the server
ID_C	Indicates the rightful owner of the ticket
AD_S	The client's perceived server address which allows the server to have many addresses
N_2	Nonce from message 2 to prevent a replay
N_3	A random value to ensure that response is fresh. It is also used to generate the session authentication code
$Times$	The value from message 2 to provide a time period of the session
$Authenticator_C$	Proves that the user knows the password and the information has not been modified
SAC	The value from message 2 is used to verify that the message is authentic

is also analyzed when the authentication exchanges are done over an insecure channel.

4.3.1 Eavesdropping Attacks

Since the authentication exchanges are performed over the HTTPS protocol, the eavesdropper cannot obtain the authentication messages. Hence, the attacker cannot learn any information. Without a secure connection, the authentication exchanges may be eavesdropped and the attacker is able to obtain messages. The attacker cannot learn the password because it is not sent to the server. However, non-secret values including the ID of the user, server's address, nonce, and times are revealed. Learning these values does not make the protocol vulnerable. The remaining values are the server's authenticator, the client's authentication, and the session authentication code. These values are generated using HMAC. As previously discussed, HMAC is secure. Therefore, the protocol is secure against eavesdropping attacks.

4.3.2 Message Replay Attacks

The SSL/TLS communication is protected against replay attacks using MAC which is computed using the secret and the sequence number. Therefore, the replayed message is detected as a duplicate. Replaying the entire session is not possible since the master secret is generated using the pre-master secret, the client and the server's

random data. Even without SSL, the protocol prevents the replay attacks. When the user has been authenticated, an attacker may be able intercept the message in Step 3 and establishes another session. Since unexpired SACs are saved, the replayed message will contain a used SAC. Therefore, the attacker's session will not be authenticated. Hence, the protocol protects against replay attacks.

4.3.3 Message Modification Attacks

For this type of attack, an attacker attempts to modify a message transmitted between the client and the server to discover the client's password or to gain unauthorized access. Modifying an SSL data stream will cause an error in the packet. The attacker will not gain knowledge of the user's password or unauthorized access. For an insecure channel, the message exchanges in Step 2 and Step 3 are protected by the authenticator and the session authentication code. Without the knowledge of the password and the server's key, the verification will fail.

4.3.4 Denial of Service Attacks

This paper limits the scope of the denial of service attacks to the level of authentication, not the underneath layers. The proposed authentication is stateless which means that the server does not need to remember any challenge values. There is no extra resource reserved for

the user. The server can securely and correctly verify the user in Step 3. The server challenge information is implicitly calculated into the server authentication code.

4.3.5 Phishing Attacks

In phishing attacks, the attackers make an attempt to obtain sensitive information such as user credentials and credit card details. Phishing is typically carried out by social engineering techniques such as email spoofing and instant messaging to deceive users. The victims receive fraudulent messages which appear to come from a trustworthy entity. The message usually directs the victim to an authentic-looking website which lures the victim to enter sensitive information. The proposed method protects the user against phishing attacks. Since the attacker does not have knowledge of the user's credentials, it cannot create a valid server authenticator. The address of the attacker is different from the address of the server. Verification will not be successful.

4.3.6 Man in the Middle Attacks

In this type of attack, the traffic between the client and the server goes through the attacker. The attacker is capable of capturing the traffic, modifying it, and replaying the modified traffic. This may be in the form of active phishing where the attacker entices the victim to enter confidential information on the impersonated website and the attacker actively modifies the information and supplies it to the server. In Step 1, the server's address will be the attacker's address. This is because the server's address is automatically obtained. The attacker will modify it to the real server's address. In Step 2, all values in clear text can be altered. However, the server authenticator and session authentication code cannot be modified without the client's secret key and the server's secret key, respectively. Therefore, when the user verifies the server authenticator, the verification process results in failure.

5 Conclusions

Password-based authentication is still widely used. However, it may be vulnerable to phishing attacks. The proposed protocol attempts to address this issue by implementing mutual authentication which both client and server must prove that they know the shared secret. A two-factor authentication which includes the knowledge factor and location factor is also used. The proposed protocol also applies a challenge-response authentication in which both server nonce and client nonce are used. This ensures that previous authentications cannot be reused in replay attacks. Moreover, the protocol utilizes timestamps to ensure exact timeliness. Finally, the server can be implemented in a stateless manner during the authentication.

In order to protect the user against phishing attacks, when the user initiates the login process, the user's secret

is dynamically derived from the username, the password, and FQDN. If the user is on the phishing site, the user's secret is generated incorrectly. As a result, the verification process would safely fail. The attacker is not able to obtain the user's credentials or perform a transaction on behalf of the user. Therefore, the proposed anti-phishing password authentication protocol automatically protects users from attackers who try to obtain the user's password or make transactions against the user's interest as illustrated in the security analysis.

Acknowledgments

This study was supported by the National Institute of Development Administration (NIDA), Thailand. The author gratefully acknowledges the anonymous reviewers for their valuable comments.

References

- [1] A. A. Ahmed and N. A. Abdullah, "Real time detection of phishing websites," in *2016 IEEE 7th Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON'16)*, pp. 1–6, Oct. 2016.
- [2] M. Štampar, "Inferential sql injection attacks," *International Journal of Network Security*, vol. 18, no. 2, pp. 316–325, 2016.
- [3] J. Chen and C. Guo, "Online detection and prevention of phishing attacks," in *Proceedings of The First International Conference on Communications and Networking in China*, pp. 1–7, Oct. 2006.
- [4] A. Y. Daeef, R. B. Ahmad, Y. Yacob, and N. Y. Phing, "Wide scope and fast websites phishing detection using urls lexical features," in *2016 3rd International Conference on Electronic Design (ICED'16)*, pp. 410–415, Aug. 2016.
- [5] S. Egelman, L. F. Cranor, and J. Hong, "You've been warned: An empirical study of the effectiveness of web browser phishing warnings," in *Proceedings of the ACM SIGCHI Conference on Human Factors in Computing Systems (CHI'08)*, pp. 1065–1074, New York, NY, USA, 2008.
- [6] M. G. Gouda, A. X. Liu, L. M. Leung, and M. A. Alam, "Spp: An anti-phishing single password protocol," *Computer Networks*, vol. 51, pp. 3715–3726, Sept. 2007.
- [7] Anti-Phishing Working Group, *Phishing Activity Trends Report*, Dec. 2016. (<http://www.antiphishing.org/resources/apwg-reports/>)
- [8] S. Gupta, A. Singhal, and A. Kapoor, "A literature survey on social engineering attacks: Phishing attack," in *Proceedings of the International Conference on Computing, Communication and Automation (ICCCA'16)*, pp. 537–540, Apr. 2016.
- [9] T. N. Jagatic, N. A. Johnson, M. Jakobsson, and F. Menczer, "Social phishing," *Communication of ACM*, vol. 50, pp. 94–100, Oct. 2007.

- [10] A. K. Jain and B. B. Gupta, "Comparative analysis of features based machine learning approaches for phishing detection," in *2016 3rd International Conference on Computing for Sustainable Global Development (INDIACom'16)*, pp. 2125–2130, Mar. 2016.
- [11] H. Krawczyk, M. Bellare, and R. Canetti, *Hmac: Keyed-hashing for Message Authentication*, Technical Report RFC 2104, Internet Engineering Task Force (IETF), Feb. 1997.
- [12] P. Kuacharoen, "A practical customer privacy protection on shared servers," in *Proceedings of the IEEE International Conference on Information Theory and Information Security*, pp. 525–529, Dec. 2010.
- [13] I. Liao, C. Lee, and M. Hwang, "A password authentication scheme over insecure networks," *Journal of Computer and System Sciences*, vol. 72, no. 4, pp. 727–740, 2006.
- [14] A. S. Martino and X. Perramon, "Phishing secrets: History, effects, and countermeasures," *International Journal of Network Security*, vol. 11, no. 3, pp. 163–171, 2010.
- [15] 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.
- [16] 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.
- [17] E. O. Osei, J. B. Hayfron-Acquah, "Cloud computing login authentication redesign," *International Journal of Electronics and Information Engineering*, vol. 1, no. 1, pp. 1–8, 2014.
- [18] B. Ross, C. Jackson, N. Miyake, D. Boneh, and J. C. Mitchell, "Stronger password authentication using browser extensions," in *Proceedings of the 14th Conference on USENIX Security Symposium (SSYM'05)*, vol. 14, pp. 2–2, Berkeley, CA, USA, 2005.
- [19] S. E. Schechter, R. Dhamija, A. Ozment, and I. Fischer, "The emperor's new security indicators," in *Proceedings of the IEEE Symposium on Security and Privacy (SP'07)*, pp. 51–65, May 2007.
- [20] R. Shirey, *Internet Security Glossary*, Technical Report RFC 2828, Internet Engineering Task Force (IETF), May 2000.
- [21] W. Stallings, *Cryptography and Network Security: Principles and Practice (7ed)*, Hoboken, NJ: Pearson, 2016.
- [22] P. Subpratatsavee and P. Kuacharoen, *Transaction Authentication Using HMAC-Based One-Time Password and QR Code*, pp. 93–98, Springer Berlin Heidelberg, Berlin, Heidelberg, 2015.
- [23] Y. Xue, Y. Li, Y. Yao, X. Zhao, J. Liu, and R. Zhang, "Phishing sites detection based on url correlation," in *2016 4th International Conference on Cloud Computing and Intelligence Systems (CCIS'16)*, pp. 244–248, Aug. 2016.
- [24] J. Youll, *Fraud Vulnerabilities in Sitekey Security at Bank of America*, July 18, 2006. (<http://cr-labs.com/publications/SiteKey-20060718.pdf>)

Biography

Pramote Kuacharoen received his B.S. and M.E. degrees in computer and systems engineering from Rensselaer Polytechnic Institute (RPI) in 1995 and 1996, respectively. He also received his M.S. and Ph.D. degrees in electrical and computer engineering from the Georgia Institute of Technology in 2001 and 2004, respectively. He joined the Department of Computer Science at National Institute of Development Administration in 2004. His research interests include computer and network security, information security, computer networks, embedded systems, and mobile applications design and development.

Role-based Access Control for Body Area Networks Using Attribute-based Encryption in Cloud Storage

Ye Tian^{1,2}, Yanbin Peng³, Gaimei Gao¹, Xinguang Peng¹

(Corresponding author: Xinguang Peng)

College of Computer Science and Technology, Taiyuan University of Technology¹

No.79, West Yingze Street, Taiyuan, Shanxi 030024, China

(Email: sxgrant@126.com)

Computer Center, Taiyuan Normal University²

No.319 DaXue Street, Yuci District Jinzhong, Shanxi 030619, China

Software Development Center, Agricultural Bank of China³

NO. 18, Lize Road, Jintang International Finance Building, Fengtai district, Beijing 100073, China

(Received Aug. 5, 2016; revised and accepted Jan. 15, 2017)

Abstract

In order to save storage space, the data collected from body area networks can be stored in a third party. However, this may bring security problems. The common method is encrypting data before outsourcing. In this paper, we design a role-based access control scheme (RACS) used in the cloud. Firstly, we classify the data which are collected from body area networks into different types, and use the ciphertext-policy attribute-based encryption to encrypt them. Secondly, we divide the ciphertext into two parts, one part is stored in cloud, and the other is in the owner. Different users own different attributes, therefore, they only can access the data when their attributes satisfy the corresponding access structure. The security of medical data is assured in this way. Thirdly, we also add the user revocation to prevent the vicious user from obtaining and modifying the data. Lastly, when the emergency happens, users can obtain the temporary key to access medical data, so as to cure the patients in the first time. We analyze the correctness, security, storage and computation overhead of the scheme. The results show that RACS can resist the ciphertext attack and superior to others in the storage space and computation overhead.

Keywords: Access Control; Attribute-based Encryption; Body Area Networks; Cloud Storage

1 Introduction

In recent years, body area networks can be used to monitor illnesses of patients. The sensors which are put in, on and around patient's body can monitor physiological

activities of patients continually, for example, the temperature, breathing, arrhythmia and endoscope. This medical treatment is very convenient for the chronic diseases and disables. The communication in body area networks has three layers, intra-BAN communications, inter-BAN communications and beyond-BAN communications [10]. "Intra-BAN communications" refers to the communications between sensors or between sensors and personal devices; "Inter-BAN communications" is the communications between personal devices and one or more access points (APs); in "Beyond-BAN communications" the authorized persons (doctors or nurses) can access medical data through Internet, so they can diagnose the patients according to their states. Database is an important part in the third layer, where the patients' personal information and medical history are stored. These data can be outsourced to the third party, such as the cloud servers. According to the access trees which are defined by the patients and the attributes users owned, users can access special medical data. Cloud is used widely for its powerful storage and convenience. Users can outsource their sensitive data in the cloud [3, 11]. If these data are stored in cloud, most of the cost can be saved. However, the cloud is outside of the owners' control, so personal information and medical data will be exposed to the third party. One of the serious challenges is protecting the confidentiality of the data [9, 14].

For the high value of medical data, the third party is always the attacks targeting of many malicious actions. It is necessary to construct a novel data access control scheme. The method of attribute-based encryption (ABE) before outsourcing is a common method to control medical data. ABE is a one-to-many encryption. If and only if

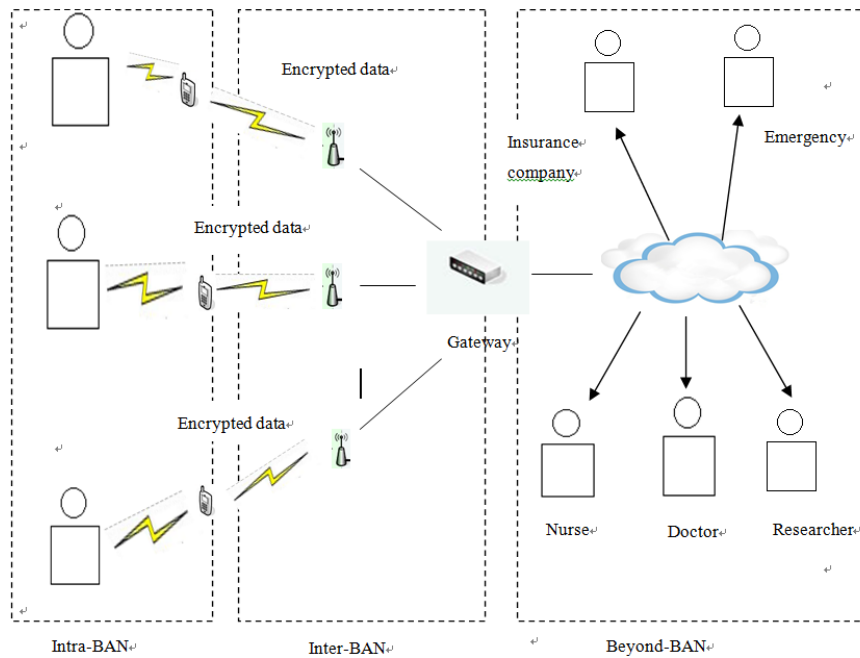


Figure 1: The three-layer architecture communication in body area networks

the attributes the users own satisfy the access tree, they can access the data. So the patients can decide the access policy. That is to say, different users can access different data. The authorized users include family members or friends, doctors, pharmacist and researchers. In the ciphertext policy (CP-ABE) schemes, the patients use access tree to encrypt data, and only when the users give the corresponding key they can access the data. CP-ABE schemes always require intensive computing resources to run the encryption and decryption algorithms. However, in body area networks the computing devices are always lightweight such as cell phones and sensors. Outsourcing the heavy computation without exposing the sensitive data contents or keys to the cloud service providers is a good choice to solve the problem.

Sahai and Waters first proposed the attribute-based encryption [12] which was built on the Identity-Based Encryption (IBE). In IBE, the users' key and ciphertext are described by strings. When the number of strings is within the threshold, the users can decrypt the ciphertext. Bethencourt and Cheung proposed the CP-ABE based on ABE [1, 2]. The key of user is associated with some attributes, and the access structure is included in ciphertext. If and only if the users' attributes satisfy the access structure, the users can decrypt ciphertext. Li and Yu focused on the multiple data owner scenario, and divided the users into multiple security domains to reduce the key management complexity [7]. Wei Li et al. conducted a threshold multi-authority CP-ABE access control scheme for public cloud storage [8]. However, the medical data can be accessed by all the users whose attributes satisfy the policy. If we want the users only can access special data, these methods are impossible, because all the data

are encrypted in one ciphertext. Wan and Wang proposed the hierarchy attribute-based encryption in cloud [13, 15]. These schemes achieve scalability and inherit flexibility and fine-grained access control, however, the medical data are not partitioned. Therefore, all the data are in the same security level. Zhou incorporated his system into mobile cloud computing scenarios. He put the intensive computation of CP-ABE encryption and decryption to cloud service providers without disclosing their data content and secret keys [16]. Therefore, the cloud services do most of the storage and computation work, alleviating the burden of the sensors. But, he didn't classify the medical data. Li et al. proposed an outsourced ABE system, which supports both secure outsourced key issuing and decryption [6]. Jung et al. proposed an AnonyControl scheme which addressed not only the data privacy but also the user identity privacy [4]. However, they also didn't consider the classification of data.

In this paper, we propose a role-based attribute-based access control scheme (RACS) by extending the CP-ABE. In this scheme, the owners classify the medical data into different parts, and encrypt them by different access policies using different attributes, in this way the users can access data according to their roles. The users can only access parts of the medical data by their roles. Therefore the security of medical data can be guaranteed. For the illegal users, the patients can revoke their access privileges. When the patients are in danger, for example, coma, the paramedics can obtain the temporary key to access the medical data to insure the efficient rescue in the short time. In order to relieve the burden of sensors, parts of the encryption and decryption are put in the cloud. We adopt a method which divides the access structure into

two parts, one part is in the sensors and the other is in the cloud.

The main contributions of this paper are: (1) we divide the medical data into different parts to realize that different users can access different parts, (2) we divide the policy into two parts: one part is stored in the cloud, and the other is in owner, (3) we consider break-glass in the scheme, (4) we analysis the correctness, security and storage and computation overhead of the scheme.

2 Preliminary

2.1 Bilinear Maps

Let G_1 and G_2 be two groups of prime order p , and g be a generator of G_1 . A bilinear map is an injective function $e : G_1 \times G_1 \rightarrow G_2$ with the following properties:

Bilinearity: $\forall u, v \in G_1, a, b \in Z_p$, there is $e(u^a, v^b) = e(u, v)^{ab}$.

Non-degeneracy: $e(g, g) \neq 1$.

Computability: There is an efficient algorithm to compute $e(u, v)$ for each $u \in G_1$ and $v \in G_1$.

2.2 Bilinear Diffie-Hellman Problem (BDHP)

Given two groups G_1 and G_2 with the same prime order p , let $e : G_1 \times G_1 \rightarrow G_2$ be a bilinear map and g be a generator of G_1 . The objective of BDHP is to compute $e(g, g)^{abc}$ in (G_1, G_2, e) from the given (g, g^a, g^b, g^c) , where $a, b, c \in Z_p$.

2.3 Access Tree

Access tree expresses the structure of access control. Let T be an access tree with root r , and $att(x)$ be the attributes associated with the node x . If num_x is the number of children of a node x and k_x is its threshold value, then $0 < num_x \leq k_x$. When $k_x = 1$, the threshold gate is an OR gate; when $k_x = num_x$, it is an AND gate. Denote T_x as the subtree of T rooted at the node x . Hence T_x is the same as T . When the attributes associated with the ciphertext satisfy the owners' access structures, the users can get the medical data. If a set of attributes satisfy the access tree T_x , we denote it as $T_x(\gamma) = 1$. $T_x(\gamma)$ can be computed recursively as follows: When x is a leaf node, $T_x(\gamma) = 1$ if and only if $att(x) = \gamma$. When x is a non-leaf node, evaluate $T_{x'}(\gamma)$ for all children x' of node x . $T_x(\gamma) = 1$ if and only if at least k_x children return 1.

3 Scheme Model

3.1 Problem Definition

We consider a role-based attribute encryption cloud storage access control scheme in which there are different own-

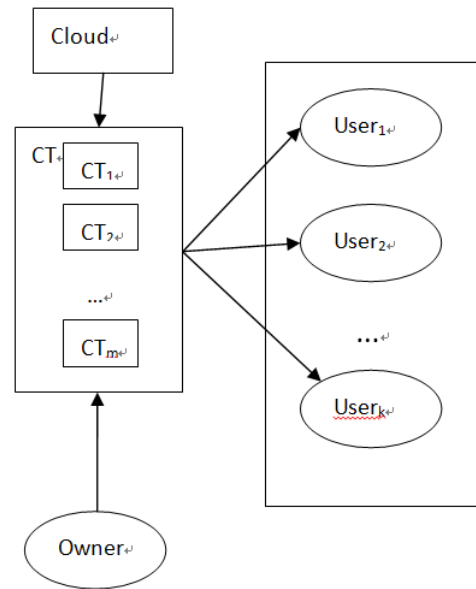


Figure 2: Storage model of medical data

ers and different users. Figure 2 shows the cloud storage model. Every patient owns his medical data and shares them to users through Internet. The medical data are stored in a third party. The patients have their control rights of medical data entirely. They can produce, manage and delete their data. The center server manages these data. The users can read or write different parts of medical data from the servers according to their attributes. When a user wants to access medical data, he first checks that which access tree his attributes satisfy. For example, if a user's attributes satisfy the access tree of CT_i , he only can obtain CT_i and can't get other parts of CT . The role-based scheme consists of the following components:

- 1) The service provider. It is the third party which controls users accessing the outsourcing data and providing outsourcing services.
- 2) Owners. The patients who own the medical data define the access policies and outsource them to the service provider after encrypting.
- 3) Center server. It is an attribute set key institution. It produces the public key and the master key. It also distributes, revokes and renews the users' private key.
- 4) Users. The persons who want to access medical data. If a user owns the attributes which satisfy the access policy, he can decrypt the corresponding data.

We show an example to illustrate the process of the scheme. Suppose Alice is a patient in hospital A. She creates her medical data file F and divides them into different parts, such as personal information, medical history, physical examination information, and sensitive data. We illustrate it in Figure 3. Alice encrypts them according

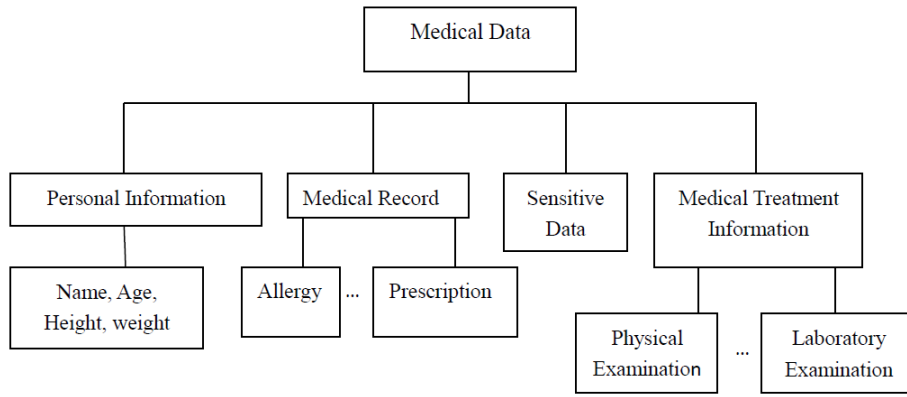


Figure 3: The division of medical data

to the access structures she defines. Different users have different access privileges as they own different attributes. For example, doctors can access all data; insurance company can only access personal information; her friends and researchers can access medical history. Alice also transport a temporary key to the trust center, when the emergency happens, the emergencies can get the key and access the fist-aid data.

3.2 Role-based Access Control Scheme in Cloud Storage (RACS)

Scheme Initialization:

- 1) Let G be the group with prime order p and g be the generator of G .
- 2) Choose two random numbers $\alpha, \beta \in Z_p$, the public key is $PK : \{g, h = g^\beta, f = g^{y\beta}, e(g, g)^\alpha\}$.
- 3) The master key is $MK : \{\beta, g^\alpha\}$.

Key Generation:

- 1) Choose a random number $\gamma \in Z_p$.
- 2) For each attribute $j \in S$, choose random numbers $\gamma_j \in Z_p$.
- 3) Generate the private key: $D = g^{(\alpha+\gamma)/\beta}, \forall j \in S : D_j = g^\gamma \cdot H(j)^{\gamma_j}, D' = g^{\gamma_j}$.

The ciphertext is divided into different parts. For each part, an access tree is constructed. Therefore, different keys are generated to decrypt them. According to users' roles, the key generation algorithm generates their private keys.

Encryption before Outsourcing:

For the patients, the privacy is an important issue. For example, a patient may don't want some users to know that he has certain diseases. We can divide the data M into N parts, $M = \{M_1, M_2, \dots, M_N\}$, and encrypt them with corresponding access trees. After encryption, the ciphertext is constructed as

$CT = \{CT_1, CT_2, \dots, CT_N\}$. $CT_k (k = 1, 2, \dots, N)$ indicates one part of ciphertext. It can be decrypted by users whose attributes satisfy the corresponding access trees. In this way, we can realize role-based access control.

In order to alleviate the heavy computation, parts of encryption and decryption are moved to cloud. An access tree is divided into two parts: $T = T_{CLOUD} \wedge T_{DO}$. T_{CLOUD} is one part of access tree which is controlled in the cloud and T_{DO} is the other part which is controlled by data owner. To relieve the computation overhead in the Inter-BAN, T_{DO} usually has a small number of attributes. Most of the computation is performed in T_{CLOUD} which is stored in cloud. We illustrate an access tree in Figure 4.

For the T_{CLOUD} , the process is as follows: Choose a polynomial q_x for each node x , and set the degree $d_x = k_x - 1$ (k_x is the threshold of x). For the root node R in the tree, choose a random number $s \in Z_p$ and set $q_R(0) = s$. For other nodes, set $q_x(0) = q_{parent(x)}(index(x))$ and choose randomly other d_x nodes to define q_x .

For the T_{DO} , the process is as follows:

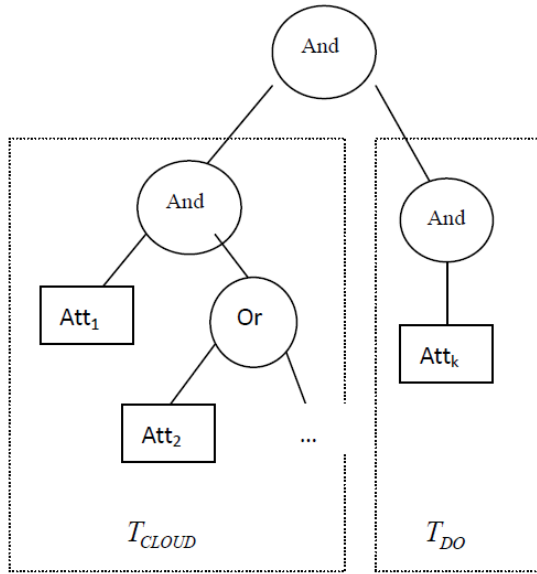
- 1) Encrypt $(q_R(2), T_{DO})$ and $CT_{DO} = \{\forall y \in Y_2 : C_y = g^{q_y(0)}, C'_y = H(att(y))^{q_y(0)}\}$.
- 2) Computer $\tilde{C} = Me(g, g)^{\alpha s}$ and $C = h^s$, where M is the message.
- 3) Send CT_{DO}, \tilde{C}, C to the cloud.

On receiving the message from data owner, the cloud server generates the following ciphertext:

$$\begin{aligned}
 CT &= \{T_{ESP} \wedge T_{DO}; \tilde{C} = Me(g, g)^{\alpha s}; C = h^s; \\
 &\forall y \in Y_{ESP} \cup Y_{DO} : C_y = g^{q_y(0)}, \\
 &C'_y = H(att(y))^{q_y(0)}\}.
 \end{aligned}$$

Data Decryption:

When a user wants to access CT_k , the center server


 Figure 4: An Access Tree of CT_k

first checks whether his attributes satisfy the corresponding access tree. If it is satisfied, the decryption is handed over to the cloud. The user sends SK_k to the cloud, and requests the cloud provider to send the ciphertext. When x is a leaf node, $i = att(x)$, the decryption process is as follows: if $i \in S$,

$$\begin{aligned} DecryptNode(CT, SK, x) &= \frac{e(D_i, C_x)}{e(D'_i, C'_x)} \\ &= \frac{e(g^r \cdot H(i)^{r_i}, h^{q_x(0)})}{e(g^{r_i}, H(i)^{q_x(0)})} \\ &= e(g, g)^{r q_x(0)} \end{aligned}$$

If $i \notin S$, $DecryptNode(CT, SK, x) = \perp$;

The recursion is processed as follows: $\forall y$ is the child of x . It calls $DecryptNode(CT, SK, y)$ and stores the output as F_y . Let S_x be an arbitrary k_x -sized set of child node y , the computation is processing in cloud as follows:

$$\begin{aligned} F_x &= \prod_{z \in S_x} F_z^{\Delta_{i, s'_x(0)}} \\ &= \prod_{z \in S_x} (e(g, g)^{r \cdot q_z(0)})^{\Delta_{i, s'_x(0)}} \\ &= \prod_{z \in S_x} (e(g, g)^{r \cdot q_{p(z)(index(z))}})^{\Delta_{i, s'_x(0)}} \\ &= \prod_{z \in S_x} e(g, g)^{r \cdot q_x(i) \cdot \Delta_{i, s'_x(0)}} \\ &= e(g, g)^{r \cdot q_x(0)} \end{aligned}$$

Where $i = index(z)$ and $S'_x = \{index(z) : z \in S_x\}$. Finally, the recursive algorithm returns $A = e(g, g)^{rs}$.

Key Update:

When users need to be revoked or attributes are changed, the owner can update the users' privileges through trusted center. Suppose there is a user revocation, the key is updated as follows:

The trust center chooses $s' \in Z_p$ randomly and a key K'_{λ_i} which is different to original K_{λ_i} , then encrypt the ciphertext again.

- 1) $C' = Me(g, g)^{\alpha(s+s')}$.
- 2) $C_i = g^{q_i(0)+s'}$, $C'_i = (H(\lambda_i)^{q_i(0)+s'})K'_{\lambda_i}$.
- 3) $\forall y \in Y/\{i\} : C_y = g^{q_y(0)+s'}$, $C'_y = (H(\lambda_y)^{q_y(0)+s'})K_{\lambda_y}$.

Break-glass:

When a patient is in emergency, the first-aiders need to access medical data temporarily. They prove their privileges from emergency response department, and get the patient's emergency key to decrypt the medical data. After the emergency treatment, the patient computes an emergency key once again.

- 1) Choose a random number $\eta \in Z_p$.
- 2) Generate the private key $D = g^{(\alpha+\eta)/\beta}$.

4 Scheme Analysis

4.1 Correctness

The decryption process starts from the root of tree. We observe $DecryptNode(CT, SK, r) = e(g, g)^{r q_R(0)} = e(g, g)^{rs}$ if and only if the attributes satisfy the access tree.

$$\begin{aligned} M' &= \frac{C'}{e((C, D)/A)} \\ &= \frac{Me(g, g)^{\alpha s}}{e(h^s, g^{(\alpha+\gamma)/\beta})/e(g, g)^{rs}} \\ &= \frac{Me(g, g)^{\alpha s}}{e(g^\beta, g^{(\alpha+\gamma)/\beta})/e(g, g)^{rs}} \\ &= \frac{Me(g, g)^{\alpha s}}{e(g, g)^{(\alpha+\gamma)}/e(g, g)^{rs}} \\ &= M. \end{aligned}$$

4.2 Security Analysis

Theorem 1. *RACS is secure against the collusion attack. In RACS, each attribute is assigned with a random number. For the users, all the private keys are generated based on these random numbers. The access tree is divided into two parts, T_{CLOUD} and T_{DO} . One is stored in cloud and the other in owners. The polynomials in the cloud is set by random numbers. Therefore, even the users collude together, they can't decrypt the part which is in cloud. Thus, it is impossible for multiple users collude together to decrypt the ciphertext.*

Table 1: Comparison of storage overhead

Scheme	Public key	Master key	Private key	Cloud storage
EDRS [5]	3	2	$3 n_i + 1$	$2 S + 3 + T $
FH-CP-ABE [15]	$3 p _1 + p _2$	$Z_p + p _1$	$(2 n_i + 1)p$	$(2S + k)p_1 + (jS + k)p_2$
RACS	4	2	$2 n_i + 1$	$2 S + 2 + T $

Table 2: Comparison of computation overhead

Scheme	Setup	KeyGen	Encrypt	Decrypt
EDRS [5]	$3E + 1e$	$(U n_i + U)M + (4 U n_i + 2 U)E$	$1M + (2 S + 2)E$	$O(S + 2)M + (2 S + 1)e$
FH-CP-ABE [15]	$2E + 1e$	$(U n_i + 1)M + (3 U n_i + 1)E$	$1M + (2 S + 2)E + 2e$	$O(S + 2)M + (2 S + 1)e$
RACS	$3E + 1e$	$(U n_i + 1)M + (U n_i + 1)E$	$1M + (2 S + 2)E$	$O(S + 2)M + (2 S + 1)e$

Theorem 2. *RACS with outsourced decryption is secure against chosen-plaintext attack in selective model under DBDH assumption.*

Proof. We now describe the security model of RACS by the following game between a challenger and an adversary.

Init: Assume there is an adversary A with attributes set W breaks the proposed scheme. We can build a simulator S that uses A as a sub-algorithm to solve the DBDH problem. The challenger S chooses a fair binary coin $\mu = \{0, 1\}$, $a, b, c \in Z_p$. If $\mu = 0$, S is given $(A, B, C, Z) = (g^a, g^b, g^c, e(g, g)^{abc})$; otherwise it sets z as a random number, S is given $(A, B, C, Z) = (g^a, g^b, g^c, e(g, g)^z)$. The challenger S runs A and receives a challenge attributes set W from A and sends public key: $PK : \{g, h = g^\beta, f = g^{1/\beta}, e(g, g)^\alpha\}$ to A .

Phase 1: In this phase, A repeatedly makes private key requests for W . The center server gives two different private keys to A .

$$SK : \quad (D = g^{(\alpha+\gamma)/\beta}, \forall j \in W : D_j = g^r \cdot H(j)^{r_j}, \\ D'_j = g^{r_j})$$

$$SK : \quad (D = g^{(\alpha+\gamma')/\beta}, \forall j \in W : D_j = g^{r'} \cdot H(j)^{r'_j}, \\ D'_j = g^{r'_j})$$

j is the attribute in W , r, r', r_j, r'_j are the random numbers in Z_p . Challenge: The adversary A submits two messages M_0, M_1 to challenger, and get the challenge ciphertext as follows:

$$CT^* = (T, C' = M_b \cdot e(g, g)^{\alpha S}, C = h_1^s = S, \\ C' = h_2^S, \forall y \in Y : C_y = g^{a_y(0)})$$

The challenger returns CT^* to A .

Phase 2: A queries the questions that is not queried in Phase 1. The challenger answers like in Phase 1.

A outputs a guess b' of b , if $b' = b$, S outputs $\mu' = 0$ to indicate that it is given a DBDH-tuple; otherwise, it outputs $\mu' = 1$ to indicate it is given a random 4-tuple.

□

4.3 Storage Analysis

We compare the RACS with other schemes according to the size of public key, private key, ciphertext for one data content. As the part of the ciphertext in RACS is stored and computed in cloud, we only consider the part which is at local. The results are shown in Table 1. Let $|p|_1$ denotes the size of an element in G_1 , $|p|_2$ denotes the size of an element in G_2 , k be the hierarchical files, N_0 be the number of owners, N_μ be the number of users, n be the number of attributes, n_i be the set of attributes belonging to user μ_i , S be the set of attributes which are used to specify the access policy in the ciphertext.

4.4 Computation Analysis

We evaluate the energy consumption on computation of RACS. The analytical results of each scheme in terms of computation are summarized in Table 2. Each of them is based on the entire computation at each phase.

5 Conclusion

In this paper, we propose a novel role-based access control scheme using CP-ABE in cloud. As the cloud servers are partially trustworthy, we don't put all the medical data to cloud. Patients have full control of their own privacy through encrypting their medical data to allow

fine-grained access. We enhance the CP-ABE so as to relieve the storage and computation overhead in body area networks. In order to protect the privacy of patients, we divide medical data into different parts. The RACS permits users to access different parts according to their roles (professional roles, qualifications, and affiliations) and greatly reduce the complexity of key management. We design RACS to encrypt the medical data. Furthermore, we add user revocation and break-glass to handle privacy and emergency problems. Through comparison, we show that our scheme is efficient than other schemes.

References

- [1] J. Bethencourt, A. Sahai, B. Waters, "Ciphertext-policy attribute-based encryption," in *Proceedings of IEEE Symposium on Security and Privacy*, pp. 321–334, California, USA, May 2007.
- [2] L. Cheung, C. Newport, "Provably secure ciphertext policy ABE," in *Proceedings of 14th ACM Conference on Computer and Communications Security*, pp. 456–465, New York, USA, Oct. 2007.
- [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] T. Jung, X. Y. Li, Z. Wan, M. Wan, "Control cloud data access privilege and anonymity with fully anonymous attribute based encryption," *IEEE Transactions on Information Forensics and Security*, vol. 10, pp. 190–199, 2015.
- [5] D. Y. Koo, J. Hur, H. Yoon, "Secure and efficient data retrieval over encrypted data using attribute-based encryption in cloud storage," *Computers and Electrical Engineering*, vol. 39, no. 1, pp. 34–46, 2013.
- [6] J. Li, X. Huang, J. Li, X. Chen, X. Yang, "Secure outsourcing attribute-based encryption with checkability," *IEEE Transactions on Parallel and Distributed Systems*, vol. 25, no. 8, pp. 2201–2210, 2014.
- [7] M. Li, S. C. Yu, Y. ZHeng, "Scalable and secure sharing of personal health records in cloud computing using attribute-based encryption," *IEEE Transaction on Parallel and Distributed Systems*, vol. 24, no. 1, pp. 131–143, 2013.
- [8] W. Li, K. Xue, Y. Xue, J. Hong, "TMACS: A robust and verifiable threshold multi-authority access control system in public cloud storage," *IEEE Transactions on Parallel and Distributed Systems*, vol. 27, no. 5, pp. 1484–1496, 2016.
- [9] C. W. Liu, W. Fu Hsien, C. C. Yang, 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.
- [10] C. Min, S. Gonzalez, V. Athanasios, "Body area networks: A survey," *Mobile Networks and Applications*, vol. 16, no. 2, pp. 171–193, 2011.
- [11] 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.
- [12] A. Sahai, B. Waters, "Fuzzy identity based encryption," in *Proceedings of 24th Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pp. 457–473, Aarhus, Denmark, May 2005.
- [13] Z. G. Wan, J. Liu, R. Deng, "HASBE: A hierarchical attribute-based solution for flexible and scalable access control in cloud computing," *IEEE Transactions on Information Forensics and Security*, vol. 7, no. 2, pp. 743–754, 2012.
- [14] Z. Wang, Y. Lu, G. Sun, "A policy-based deduplication mechanism for securing cloud storage," *International Journal of Electronics and Information Engineering*, vol. 2, no. 2, pp. 70–79, 2015.
- [15] S. Wang, J. Zhou, J. K. Liu, J. Yu, J. Chen, W. Xie, "An efficient file hierarchy attribute-based encryption scheme in cloud computing," *IEEE Transactions on Information Forensics and Security*, vol. 11, no. 6, pp. 1265–1277, 2016.
- [16] Z. B. Zhou, *On Efficient and Scalable Attribute Based Security Systems*, Arizona State University, Arizona, 2011.

Biography

Ye Tian received her M.E degree from Taiyuan university of technology, China, in 2006. She is currently a ph.D.student in Taiyuan university of technology and an associate professor in Taiyuan Normal University. Her current research interests are information security and wireless body area networks.

Yanbin Peng received the Master degree in Engineering in computer technology in 2016 from the Taiyuan University of Technology, Taiyuan, China. She is currently a software engineer at Software Development Center, Agricultural Bank of China. Her research interests include network and information security.

Gaimei Gao received the M.E from Taiyuan University of Science and technology, China, in 2007. She is currently a ph.D.student in Taiyuan university of technology. Her research interests are Database and Information Security.

Xinguang Peng received the D.E from Beijing Institute of technology, China in 2004. He is now a professor and doctoral supervisor in college of computer science and technology, Taiyuan University of Technology, Taiyuan, China. His research interests include information security and trusted computing.

A Comparative Study on Feature Selection Method for N-gram Mobile Malware Detection

Mohd Zaki Mas'ud, Shahrin Sahib, Mohd Faizal Abdollah, Siti Rahayu Selamat, Choo Yun Huoy
(Corresponding author: Mohd Zaki Mas'ud)

Faculty of Information Technology and Communication, Universiti Teknikal Malaysia Melaka
Hang Tuah Jaya, 76100 Durian Tunggal, Melaka, Malaysia
(Email: zaki.masud@utem.edu.my)

(Received Feb. 18, 2016; revised and accepted May 21 & July 19, 2016)

Abstract

In recent years, mobile device technology has become an important necessity in our community at large. The ability of the mobile technology today has become more similar to its desktop environment. Despite the advancement of the mobile devices technology provide, it has also exposes the mobile devices to the similar threat it predecessor possess. One of the anomaly based detection methods used in detecting mobile malware is the n-gram system call sequence. However, with the limited storage, memory and CPU processing power, mobile devices that provide this approach can exhaust the mobile device resources. This is due to the huge amount of system call to be collected and processed for the detection approach. To overcome the issues, this paper investigates the use of several different feature selection methods in optimizing the n-gram system call sequence feature in classifying benign and malicious mobile application. Several filter and wrapper feature selection methods are selected and their performance analyzed. The feature selection methods are evaluated based on the number of feature selected and the contribution it made to improve the True Positive Rate (TPR), False Positive Rate (FPR) and Accuracy of the Linear-SVM classifier in classifying benign and malicious mobile malware application.

Keywords: Feature Selection; Linear SVM; Mobile Malware; Mobile Malware Detection; N-gram

1 Introduction

The number of mobile malware has increased significantly within these recent years especially with the introduction of the android-based platform in the market. Android-based mobile device offers credibility; performance and ease of customizing has made it a preferable choice for most of mobile device users. Consequently, the high reputation of android-based mobile devices has invited the malware author to make it as a new target to exploit.

This is shown in the 2013 Kaspersky's Lab report which reveals 98% of the mobile malware found in 2013 is targeting the Android platform [11]. Additionally, in 2015 new samples of mobile malware are still continuing to increase, this is based on the report done by the 2015 McAfee Labs Threats Report [20]. In order to overcome these issues, several researches had been done in finding the defense mechanism against the android-based mobile malware.

Previous research done in mobile malware detection showed that mobile malware detection can be classified into 3 different techniques which are signature-based, anomaly-based and specification-based [19]. Signature-based approach detects malware by comparing the mobile application activity signature with the database of known attack or threat. Even though it has been used in developing most of the antivirus software and successfully detects known malware with a high accuracy, the technique has a drawback in detecting unknown malware. On the other hand, the anomaly-based and specification-based detection techniques have - great reputation in detecting unknown or new malware but these two techniques tend to generate false alert or generating misclassification. Using the advantage of anomaly-based detection technique, this research has applied n-gram system call sequence as the feature to improve the classification accuracy and reduce the false alert. However, the n-gram system call sequence generates a huge number of features that can increase the processing time and complexity. Thus, in order to reduce the number of features and at the same time improves the classification accuracy, as well as minimizing the false alert, this paper investigates several existing feature selection approaches.

The aim of this research is to find the feature selection method that can provide an optimum n-gram system call sequence feature to be used in the classification of benign and malicious mobile application. Feature selection is one of the essential techniques in data mining especially during the data processing [3, 16]. The main objective of feature selection phase is to improve the mining performance as well as improving the detection accuracy by removing

irrelevant, redundant and noisy data from the dataset. Subsequently, as the number of irrelevant features is removed the data mining process become less complex to process and this can speed up the classification or clustering process.

The remainder of this paper is divided into four parts whereby section two and three review the background domain of n-gram system call sequence in mobile malware detection and the related feature selection approach investigated in this research. Section four explains the experimental setup used in evaluating the selection feature selection method. Section five presents and discusses the experimental result. Finally, the conclusion and future work is drawn in the last section.

1.1 N-gram in Mobile Malware Detection

Mobile malware has become a lethal threat to mobile device users as the effects of mobile malware infection can be from stealing confidential information from the device, monitoring user activity and location, overcharge users by sending random SMS and MMS to contact, launching denial of services attack from user devices and overloading device resources such as memory, battery and storage [17]. One of the options in detecting these activities on mobile devices is by monitoring system call invoked by the mobile application [18]. Xi et al. [27], Crowdroid [5], Isohara et al. [12], AMDA [1] and MADAM [9] are among the works using system call as the features in classifying benign and malicious mobile application. From all these - works, only Isohara et al. use signature-based detection approach and the rest use anomaly-based detection approach that takes each single occurrence of the system call as the feature.

The use of anomaly-based detection approach in detecting mobile malware application can lead to the generation of false alert in which the benign application might be misclassify as a malicious mobile application or the other way around. This issue can be improved by using a feature of a sequence system call occurrence which has been used text classification and speech recognition domain [7, 13, 22, 24]. Known as n-gram analysis, n is the value of the number of sequence and it can represent the whole system call invoke to execute a malicious. For malware detection, n-gram analysis approach has been implemented in classifying malware using its byte level information [4, 21] and its opcode [6, 26] but this requires the malware application to be decompiled before the detection process take places.

Despite the improvement in reducing the false alert, the n-gram analysis can cause a huge number of features to be captured and processed. This is not an applicable option in a limited processing power and resources devices such as mobile device. The number of system call in an android 4.0.4 OS is 300 [25], yet only 111 system call is invoked during the experiment. Accordingly, as the n value increases, the total number of system call sequence used as features is also increases to the power of n. For example, for n=2 the total of system call sequence to be

collected for this experiment is equal to 1112 or 12 321 and if n=3, the total of system call sequence to be collected is equal to 1113 or 1 367 631 which is quite a huge number to be processed in a classification problem. To reduce the number of relevance features in the classification, this research investigates several feature selection methods. The best feature selection method is evaluated based on the optimum number of features it generates and how good the features contribute in improving the classification accuracy while reducing the false alert.

1.2 Feature Selection

The n-gram system call sequence can generate a large number of features to be used in the classification and can contribute to the degradation of classification performance. This can be caused by the existence of useless features that might not be useful at all in classifying the problem. In order to overcome this matter, feature selection method is introduced in the framework for the purpose of finding the optimum features which can improve the classification performance and accuracy [2, 8, 10, 23]. In addition, the feature selection also contributes in reducing the number of selected features to be logged, thus less number of storage is used.

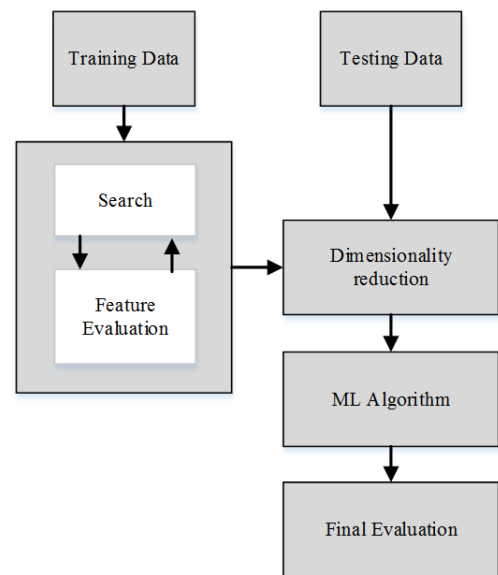


Figure 1: Filter method

Generally, there are 3 feature selection methods; filter method, wrapper method, and embedded method [23]. Filter method is illustrated in Figure 1. This method evaluates the significance of each feature using statistical approach that scored and ranked most relevance features. Features that obtained the highest ranked and scored are most likely to be chosen as the features in machine learning problem, whereas features with the lowest value are removed. Filter method is fast to compute and not affecting any of the classifier used; however this method ignores the feature dependencies and disregards the in-

teraction with the classifier. Furthermore, the threshold or the cut off value of the feature scored and ranked is not properly specified.

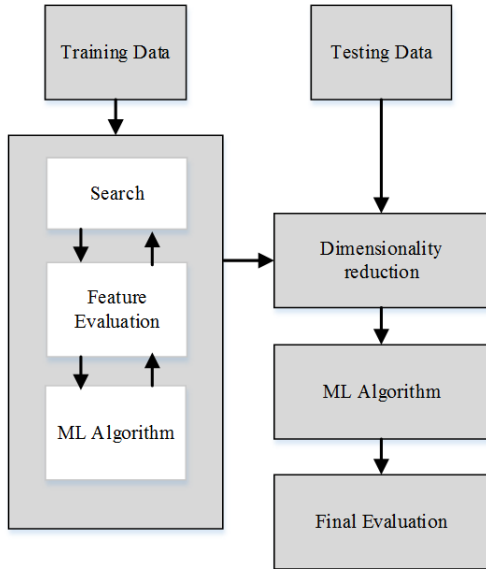


Figure 2: Wrapper method

Figure 2 illustrates wrapper method which evaluates subset of features using induction algorithm that incorporated the classifier as part of the evaluation. Thus, the features selected are specifically tailored and optimize to the classifier. Even though it is computationally intensive, wrapper method provides a possibility of interaction between features that can generate a more accurate classification. Meanwhile, embedded method is proposed to incorporate the advantages of the filter and wrapper method. The features are evaluated inside the induction algorithm itself and computationally intensive compared with wrapper methods. Nevertheless, for the purpose of finding the optimum selection method in the n-gram system call sequence feature, this paper only evaluates filter and wrapper selection methods. Four different filter methods, namely Correlation-based Feature Selection (CFS), Chi Square (CHI), Information Gain (IG), Relief (RF) and one wrapper method with a Linear SVM classifier (WR) are chosen to be evaluated in this paper.

CFS selects feature subset based on the maximal correlation of the subset to the class and the minimal correlation between the features. The features are ranked by using a correlation based heuristic evaluation function [14]. Meanwhile, CHI method evaluates feature subset with respect to the class labels based on the χ^2 -statistic function. The features are ranked and the higher the features ranked, the most likely it is chosen as the features. Similarly, IG also select features by ranking the feature based on the score generated on how much information about the class is gained when using the feature. RF assesses an attribute by repetitively sampling a feature and considering the value of the given attribute for the nearest features of the same and different class [14]. The wrapper

method generates a subset of feature candidate using a search method and applied it to the Linear SVM classifier to be evaluated using the classification Accuracy and Root Mean Square Error (RMSE) [15]. The feature subset that produced the best accuracy and RMSE is used as the features. The next section describes the methodology and the experimental setup used in evaluating these feature selection methods.

2 Methodology

The objective of this study is to compare and suggest feature selection method to be used in selecting the optimum system call features in classifying benign and malicious android application. To achieve the objective stated, an extensive and rigorous empirical comparative study is designed and conducted. The experiment is conducted through several phases namely system call log phase, n-gram extraction phase, feature selection method comparison phase and followed by machine learning classifiers phase that is used for evaluating the feature selection method. The entire phase involved in this study is illustrated in Figure 3.

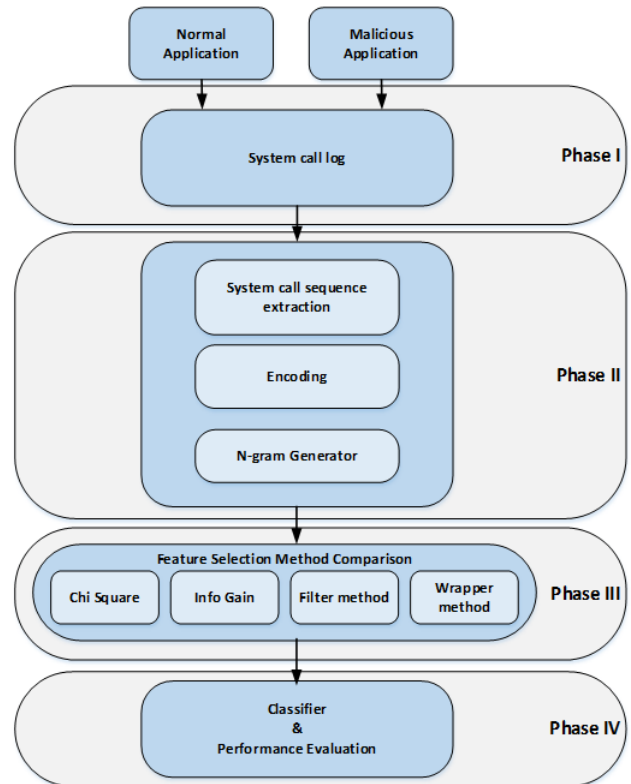


Figure 3: Research methodology

The four phases in the research methodology in Figure 3 begins with data collection phase where the system call invoked by the application is collected. Mobile applications used in the experiment are from 102 malware infected applications from the MalGenome Project [28] and 100 normal android application downloaded from Google

Play. In order to validate the benign and malicious application, the application used in this experiment has been verified by Bitdefender, eseT and VirusTotal for verification whether it is truly a malicious or benign application. Each android application is executed on a Samsung P6800 Galaxy tab 7.7 that is connected to a network experimental testbed via Wi-Fi. Each tablet is also provided with an active GSM service. Each application installed in the tablet is stimulated with user interaction as well as other mobile users common activities such as web browsing, sending and receiving SMS for duration of 10 minutes. During the execution and stimulation processes, a tool called strace is used to capture all the system calls invoked by the installed application. Once the android application has gone through these processes, the tablet is wiped out clean to its factory setting before the next application is executed.

The captured system call is then processed in the second phase where the output from the strace is transformed to a sequence of n-gram system call. Then in the third phase, the system call sequence is applied to several feature selection methods for generating the best feature. The next phase takes the best feature generated from each feature selection method and applied it to the classification method. This final phase also evaluated the number of feature selected and the classification performance in term of the Accuracy, True Positive Rate (TPR) and False Positive Rate (FPR).

3 Analysis and Discussion

The primary consideration in this study is to evaluate feature selection method that can reduce the number of features selected while improving the classification between the benign and malicious mobile application. The n value for the n-gram system call sequence used in this experiment is 3. This is based on the reason that if all the features are used during the classification, the optimum performance of the classification is only achieved when the n value of the n-gram system call sequence is 3. Table 1 shows the classification performance evaluation done on the dataset using all the features for n value of 1, 2, 3, 4, 5 and 6.

Table 1 shows all the classification performance evaluation results when all features are considered for each n-gram system call sequence. The highest accuracy in the classifier performance evaluation result is 96.19%, achieved when the n value is 3. Even though the TPR value is not the highest, yet the overall classification accuracy and FPR value is still higher than the other N-gram system call sequence. The results also show that the higher the number of sequence considered in generating the features does not affect the classification, instead it can produce a sparse vector resulting in lower accuracy. This caused by the existence of useless features that might not be useful at all in classifying the problem.

Despite of the higher classification accuracy produced

by the 3-gram system call sequence, the 3-gram still use a large number of feature which is 41142. This large number of feature can degrade the classification performance. To improve the classification performance, each feature selection method discussed earlier is then applied to the 3-gram dataset for finding the most relevant feature and at the same time reducing the number of features. The result of each feature selection method is shown in Table 2.

Table 2 shows the number of features selected by each method with the TPR, FPR and Accuracy. Out of 10 methods, only WR+ES method reduced the number of features less than 50%. Six methods which are CFS+ES, CFS+GS, CFS+PSO, CHI+Ranker, IG+Ranker, RF+Ranker, WR+GA and WR+PSO successfully reduced the features to more than 50%. Two methods are able to reduce the features up to 99%, CFS+BFS reduce the features to only 83 features whereas WR+BF can reduce the features up to only 10.

The second measurement of this study is to measure the improvement of Accuracy, TPR and FPR when the features selected are applied to the classification method. The evaluation shows CFS+BF, WR+BF, WR+ES, WR+GA and WR+PSO have improved the classification accuracy to more than the original classification accuracy which is 96.2%. WR+BF selection and search methods improved the TPR, FPR and accuracy to 100%, 2% and 99% respectively making it the best features selection method for this evaluation. Although the result shows WR+BF has the smallest number of features selected, this selection method have better ability to support the classifier to accurately classify between the malicious and benign application compare to the other features suggested by the other selection methods. This shows that the WR+BF have the ability to find an optimum features that is optimally design for the classifier.

4 Conclusion

The rapid evolution in mobile device technology has triggered a sudden increase of mobile malware threat. The effects of mobile malware threat are devastating especially when communities nowadays are depending on mobile device to store crucial information. An anomaly-based detection using n-gram system call sequence is one option that can be used to mitigate the malicious application from exploiting vulnerabilities in mobile device. However, the approach can create a large number of features are as the n value increases and can degrade the classification performance. Based on this reason, this paper evaluates several feature selection methods by comparatively analyze the performance of each selection method. Each selection method is evaluated based on the number of feature selected and the contribution it made to improves the True Positive Rate (TPR), False Positive Rate (FPR) and Accuracy of the Linear-SVM classifier in classifying benign and malicious mobile

Table 1: The classifier performance evaluation result

N-gram Dataset	Number of features	TPR (%)	FPR (%)	Accuracy (%)
1-gram	111	96.07	78	87.08
2-gram	3631	96.07	91	93.51
3-gram	41142	97.06	5	96.19
4-gram	186610	100	85	92.33
5-gram	491782	100	71	85.59
6-gram	987263	100	58	79.16

Table 2: The feature selection method performance evaluation result

Feature Selection Method	Search Method	Number of Features Selected	Reduce Percentage	TPR (%)	FPR (%)	Accuracy (%)
None		41142		98.0	6.0	96.2
CFS	BF	83	99.80	99.0	4.0	97.5
	ES	12872	68.71	96.1	6.0	95.1
	GS	10950	73.38	96.1	12.0	92.1
	PSO	10495	74.49	96.1	14.0	91.1
CHI(50%)	Ranker	20572	50.00	98.0	6.0	96.0
IG(50%)	Ranker	20572	50.00	98.0	6.0	96.0
RF(50%)	Ranker	20572	50.00	99.0	6.0	95.0
WR	BF	10	99.98	100.0	2.0	99.0
	ES	23773	42.22	98.0	3.0	97.5
	GA	17490	57.49	99.0	3.0	98.0
	PSO	16874	58.99	99.0	3.0	98.0

malware application. The selection method evaluated in this paper are Correlation-based Feature Selection (CFS), Chi Square (CHI), Information Gain (IG), ReliefF (RF) and wrapper (WR) method with a Linear SVM classifier (WR). CFS and wrapper evaluator are match with four search method namely BestFirst (BF), Evolutionary Search (ES), Genetic Search (GS) and Particle Swarm Optimization (PSO) search whereas CHI, IG and RF are match with Ranker method. Each feature generated by the selection method is then applied to a Linear-SVM classifier for TPR, FPR and Accuracy. The evaluation shows an increase in accuracy for all the features generated by the feature selection algorithm and WR-BF generated the smallest number of feature while having an accuracy of 99% and small FPR of 2%. This shows that WR+BF have the ability to find and optimum features to be used in the classifier. Moreover, the result also shows that it is possible to improve the detection performance even though the features selection used in the classifier is reduced. This small number of feature can help reduce the size of log collection in the mobile device. In the near future, the method presented in this paper may be potentially applied to develop an android malware detection that can address the limitation and constrain of mobile devices environment especially on the storage, memory and power consumption usage.

Acknowledgments

The authors would like to thank INSFORNET Research Group of Universiti Teknikal Malaysia Melaka (UTeM) for the financial support under the Fundamental Research Grant Scheme with Project No. FRGS/1/2015/ICT04/UTeM/02/F00290.

References

- [1] K. J. Abela, J. R. D. Alas, D. K. Angeles, R. J. Tolentino, and M. A. Gomez, "Automated malware detection for android: Amda," in *The Second International Conference on Cyber Security, Cyber Peacefare and Digital Forensic (CyberSec'13)*, pp. 180–188, 2013.
- [2] B. Arslan, S. Gunduz, and S. Sagiroglu, "A review on mobile threats and machine learning based detection approaches," in *4th IEEE International Symposium on Digital Forensic and Security (ISDFS'16)*, pp. 7–13, 2016.
- [3] A. L. Blum and P. Langley, "Selection of relevant features and examples in machine learning," *Artificial Intelligence*, no. 97, pp. 245–271, 1997.
- [4] T. B. Assaleh, V. Keselj, and R. Sweidan, "N-gram based detection of new malicious code," in *Proceedings of The 28th IEEE Annual International Computer Software and Applications*, pp. 41–42, Hong Kong, Sept. 2004.

- [5] I. Burguera, U. Zurutuza, and S. Nadjm-Tehrani, "Crowdroid: Behavior-based malware detection system for android," in *Proceedings of the 1st ACM Workshop on Security and Privacy in Smartphones and Mobile Devices*, pp. 15–26, 2011.
- [6] G. Canfora, A. D. Lorenzo, E. Medvet, F. Mercaldo, and C. A. Visaggio, "Effectiveness of opcode n-grams for detection of multi family android malware," in *10th IEEE International Conference on Availability, Reliability and Security (ARES'15)*, pp. 333–340, 2015.
- [7] W. B. Cavnar and M. T. John, "N-gram-based text categorization," *Ann Arbor MI 48113.2*, vol. 48113, no. 2, pp. 161–175, 1994.
- [8] M. Dash and H. Liu, "Feature selection for classification: Intelligent data analysis," *Intelligent Data Analysis*, vol. 1, no. 1, pp. 131–156, 1997.
- [9] G. Dini, F. Martinelli, A. Saracino, and D. Sgan-durra, "Madam: A multi-level anomaly detector for android malware," in *International Conference on Mathematical Methods, Models, and Architectures for Computer Network Security*, pp. 240–253, 2012.
- [10] F. Fernández-Gutiérrez, J. I. Kennedy, S. Zhou, R. Cooksey, M. Atkinson, and S. Brophy, "Comparing feature selection methods for high-dimensional imbalanced data: Identifying rheumatoid arthritis cohorts from routine data," in *IEEE International Conference on Industrial Engineering and Systems Management (IESM'15)*, pp. 236–241, 2015.
- [11] C. Funk and M. Garnaeva, *Kaspersky Security Bulletin 2013. Overall Statistics for 2013*, Technical Report, Kaspersky, 2013.
- [12] T. Isohara, T. Keisuke, and K. Ayumu, "Kernel-based behavior analysis for android malware detection," in *Seventh IEEE International Conference on Computational Intelligence and Security (CIS'11)*, pp. 1011–1015, 2011.
- [13] D. Jurafsky and H. M. James, *Speech & Language Processing*, India: Pearson Education, 2000.
- [14] K. Kira and A. R. Larry, "A practical approach to feature selection," in *Proceedings of The Ninth International Workshop on Machine Learning*, pp. 249–256, Scotland, UK, July 1992.
- [15] R. Kohavi and H. J. George, "Wrappers for feature subset selection," *Artificial Intelligence*, vol. 97, no. 1, pp. 273–324, 1997.
- [16] H. Liu and H. Motoda, *Feature Extraction, Construction and Selection: A Data Mining Perspective (2nd printing)*, Boston: Kluwer Academic Publishers, 2001.
- [17] L. P. Mariantonietta, F. Martinelli, and D. Sgan-durra, "A survey on security f or mobile devices," *IEEE Communications Surveys & Tutorials*, vol. 15, no. 1, pp. 446–471, 2013.
- [18] M. Z. Mas'ud, S. Sahib, M. F. Abdollah, S. R. Selamat, R. Yusof, and R. Ahmad, "Profiling mobile malware behaviour through hybrid malware analysis approach," in *9th International Conference on Information Assurance and Security (IAS'13)*, pp. 78–84, 2013.
- [19] M. Z. Masud, S. Sahib, M. F. Abdollah, S. R. Selamat, and R. Yusof, "Android malware detection system classification," *Research Journal of Information Technology*, vol. 6, no. 4, pp. 325–341, 2014.
- [20] McAfee, *McAfee Labs Threats Report 2015*, Technical Report, McAfee, 2015.
- [21] R. Moskovitch, D. Stopel, C. Feher, N. Nissim, N. Japkowicz, and Y. Elovici, "Unknown malcode detection and the imbalance problem," *Journal in Computer Virology*, vol. 5, no. 4, pp. 295–308, 2009.
- [22] S. Nagaprasad, T. R. Reddy, P. V. Reddy, A. V. Babu, and B. VishnuVardhan. "Empirical evaluations using character and word n-grams on authorship attribution for telugu text,". in *Intelligent Computing and Applications*, pp. 613–623, 2015.
- [23] S. F. Pratama, A. K. Muda, Y. H. Choo, and N. A. Muda, "A comparative study of feature selection methods for authorship invarianceness in writer identification," *International Journal of Computer Information Systems and Industrial Management Applications*, vol. 4, pp. 467–476, 2012.
- [24] M. Ren and S. Kang, "Document classification using n-gram and word semantic similarity," *International Journal of u-and e-Service, Science and Technology*, vol. 8, no. 8, pp. 111–118, 2015.
- [25] T. Robot, *Android Platform Bionic*, 2014. (https://github.com/android/platform_bionic/blob/master/libc/SYSCALLS.TXT)
- [26] A. Shabtai, M. Robert, F. Clint, D. Shlomi, and E. Yuval, "Detecting unknown malicious code by applying classification techniques on opcode patterns," *Security Informatics*, vol. 1, no. 1, pp. 1–22, 2012.
- [27] X. Xi, F. Peng, X. Xianni, J. Yong, L. Qing, and L. Runiu, "Two effective methods to detect mobile malware," in *4th International Conference on Computer Science and Network Technology (ICC-SNT'15)*, vol. 1, pp. 1041–1045, 2015.
- [28] Y. Zhou and J. Xuxian, "Dissecting android malware: Characterization and evolution," in *Proceedings of The IEEE Symposium on Security and Privacy (SP'12)*, pp. 95–109, San Francisco, California, May 2012.

Biography

Mohd Zaki Mas'ud is a lecturer at the Universiti Teknikal Malaysia Melaka, Malaysia and current pursuing his PhD study in Malware Analysis. His research interest include network forensic, cyber terrorism, intrusion detection, network security , network management and penetration testing.

Shahrin Sahib received the Bachelor of Science in Engineering, Computer Systems and Master of Science in Engineering, System Software in Purdue University

in 1989 and 1991 respectively. He received the Doctor of Philosophy, Parallel Processing from University of Sheffield in 1995. He is a professor and the Vice Chancellor of Universiti Teknikal Malaysia Melaka. His research interests include network security, computer system security, network administration and network design. He is a member panel of Experts National ICT Security and Emergency Response Center and also Member of Technical Working Group: Policy and Implementation Plan, National Open Source Policy.

Mohd Faizal Abdollah is currently an associate professor at the Universiti Teknikal Malaysia Melaka, Malaysia. He received his Doctor of Philosophy in Computer Science. His research interests include network forensic, cyber terrorism, intrusion detection, network security, network management and penetration testing.

Siti Rahayu Selamat is currently a lecturer at the Universiti Teknikal Malaysia Melaka, Malaysia. She received her Doctor of Philosophy in Computer Science. Her research interests include network forensic, cyber terrorism, intrusion detection, network security and penetration testing.

Yun-Huoy Choo was born in Johor, Malaysia, in 1977. She received the B.Sc. and M.Sc. degree from the University of Technology Malaysia, in 2000 and 2002, respectively. In 2008, she was awarded the PhD in System Management and Science from the National University of Malaysia specializing in Data Mining. Since June 2002, she has been with the Faculty of Information and Communication Technology, Universiti Teknikal Malaysia Melaka (UTeM), Malaysia, where she was a Lecturer, became a Senior Lecturer in 2009, and an Associate Professor in 2015. Her current research interests include the fundamental studies of rough set theory, fuzzy sets theory, association rules mining, and feature selection, besides the application of data science and data mining in different domains includes person authentication using bio signal, muscle endurance analysis, machine failure analysis, personalized itinerary and route planning, etc.

Discriminating Flash Events from DDoS Attacks: A Comprehensive Review

Sunny Behal¹, Krishan Kumar², Monika Sachdeva¹

(Corresponding author: Sunny Behal)

I. K. Gujral Punjab Technical University¹

Kapurthala, Punjab 144603, India

(Email: sunnybehal@sbsstc.ac.in)

Information Technology Department, University Institute of Engineering and Technology²

Chandigarh, India

(Received June 23, 2016; revised and accepted Sept. 3 & Nov. 15, 2016)

Abstract

Millions of people across the globe access Internet-based applications and web services in their day to day activities. Distributed Denial of Service (DDoS) attack is one of the prominent attacks that cripple down the computing and communication resources of a web server hosting these services and applications. The situation turns further crucial when DDoS attacks are launch during similar looking legitimate traffic called a flash event (FE). Both DDoS attacks and FEs causes a sudden surge in the network traffic leading to delay in the responses from the web server. It often leads to massive financial losses, and thus, require timely actions. This paper presents a comprehensive review that broadly discusses the DDoS and FE problem, and recapitulates the recently published strategies in this field. As part of the work, a pragmatic list of rationales to discriminate the two has been proposed. This list can help the researcher community for better understanding the problem and can provide more effective solutions to the ongoing problem of discriminating DDoS attacks from FEs.

Keywords: DDoS Attack; Discrimination; Flash Event

1 Introduction

A DDoS attack deploys the collection of compromised hosts and results in unavailability of network resources for the intended users. Not directly or permanently damaging the data, but intentionally compromising the availability of the resources is the motive of these attacks [24]. However, the attackers keep on strengthening their proficiency for launching sophisticated DDoS attacks by compromising the freely available credulous hosts. Differentiating DDoS attacks from legitimate traffic is an immense chal-

lenge to the network security researchers since the attackers strike with more suave techniques to the victim every time. Almost all types of DDoS attacks are launched using botnets nowadays [13]. The prominent websites are the prime victims of such DDoS attacks. Recently Twitter, Spotify, and Amazon suffer interruptions in their services for almost two hours on Oct 21, 2016, because of DDoS attacks. Such interruptions in the services lead to huge financial losses. The revenue loss has amplified to \$209 million in the first quarter of 2016, compared to \$24 million for all of 2015 [8]. According to the recent Worldwide Infrastructure Security Report (WISR), the traffic volume of such attacks has amplified to around 600 Gbps in the year 2015 [14].

Apart from detecting of DDoS attacks, there is an another kind of network traffic which is gaining popularity among security researchers, and which causes a denial of service to legitimate users of a web service, is a Flash Event (FE). As per [4], an FE is similar to high-rate DDoS (HR-DDoS) attack wherein thousands of legitimate users try to access a particular computing resource such as a website simultaneously. This sudden surge in legitimate traffic is mainly due to some breaking news happening around the world like the publishing of Olympic schedule or new product launch by companies like Apple, Samsung, etc. It causes the untimely delivery of responses from web service and thus, require immediate action. As there are only a few parametric differences between DDoS attacks and FE traffic, it is very challenging to discriminate the two [6]. The typical network traffic profile of a DDoS attack and an FE is shown in Figure 1(a) and Figure 1(b) respectively.

In this paper, we have presented a comprehensive review of the recent solutions proposed by the fellow researchers to discriminate DDoS attacks from similar looking FEs. We have compared the existing work on a set of

identified attributes. A list of distinct detection metrics and rationales is also provided which has been used to prominently to discriminate the two types of traffic.

The rest of the paper is organized as follows: The Section-2 present the recent flash events, Section-3 describe the review of existing countermeasures for discriminating DDoS attacks from FEs. The Section-4 summarizes the core rationales that can discriminate the two, Section-5 highlights the key research gaps in the existing research, and finally, the last section conclude the paper by highlighting the scope for future work.

2 Recent Flash Events

Many FEs have occurred in recent times which have lead to the untimely responses to the legitimate users. Some of the famous examples of FEs are:

- In August 2016, millions of users simultaneously accessed the Australian census website to fill their personnel details. The lack of sufficient resources on the web server causes the website to crash down [7].
- In February 2016, a new phone was launched with a lowest ever price of INR 251 named as freedom251. It attracted millions of people in a short span of time and lead to the crash down of the web server in few hours.
- In November 2014, the announcements of attractive schemes by leading online shopping vendors like Amazon, Flipkart, Snapdeal, etc. resulted in the shutdown of their shopping website for about an hour.
- In June 2014, a unique breakdown occurred at Microsoft office, when their products like Exchange & Lync, MS Office 360 were not available online. The leading traffic peaks overwhelmed the huge amount of network elements, which results in unavailability of the functionality of Lync for a longer time.
- In September 2013, the launch of iOS7 update by Apple lead to a surge in network traffic from 1.4 Gbps to 6 Gbps when thousands of students at various universities in US began to download it simultaneously.
- In November 2012, an online shopping initiative in Australia clickfrenzy.com.au suffered unexpected surge in the network traffic leading to a dramatic reduction in the response of web server and the website failed within minutes of its launch.
- In October 2012, the news of Sandy storm in the USA result in surge of Internet traffic to around 150% in few hours [4].
- In June 2012, George Takei (the Star Trek hero) broadcast a link about the selling of 'Takei T-shirts' on his Facebook page [4]. This post engages around

2 millions of his fans directing them to a web server having limited resources in no time. High hit rate forces the website to shut down for several hours.

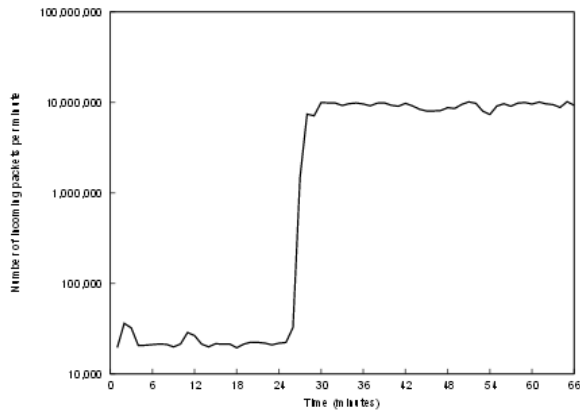
- From 11 June to 11 July 2012, the Twitter website suffered four times increase in tweets per second than an average day due to soccer world-cup news [4].
- In Oct. 2011, the death announcement of Apple's co-founder Steve Jobs result in a surge in the number of hits on his Twitter account, to around 42,000 tweets per second The news websites such as CNN and the Washington Post also experienced slowdowns of their mobile sites as people fascinated to get more information about him.
- In Feb. 1999, the victoria's secret webcast [16] of their first annual online fashion show attract around 1.5 million visitors in a short span of time which lead to dramatic increase in traffic to the host server.
- From 1 May to 24 July 1998, the FIFA world cup website remains overloaded due to the publishing of soccer world-cup event schedule and experience massive increase in web traffic from day 45 to 80 of the event [4].

3 Review of Existing Work

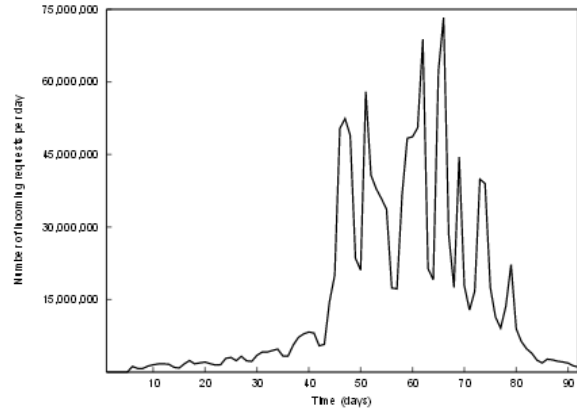
In this section, we have summarized the recent work done in the field of differentiating HR-DDoS attacks from FEs as shown in Table 1 and Table 2. The primary motive of this summary is to highlight the several rationales and detection metrics that have been used by the fellow researchers in recent times. We have compared the existing work on a set of common attributes like the type of packet header features used for detection and discrimination, detection metric, validation technique and datasets used.

Jung et al. [10] proposed a set of fundamental parameters to discriminate a DDoS attack from an FE. They analyzed the HTTP traces compiled from two engaged web servers, one from Playalong website and other from Chile website; and the log files of Code Red worm to propose parameters to distinguish an FE from a DDoS attack. They observed that the request rate per client is more in a DDoS attack than in an FE. The cluster overlapping in an FE is around 42.7% - 82.9% as compared to a DDoS attack, where it is around 0.6%-14%. It means that in the case of an FE, most of the clients have already visited the website earlier, whereas, in a DDoS attack, most of the clients are new.

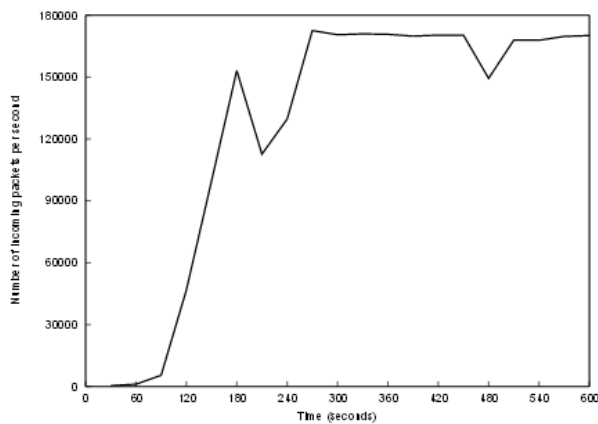
They observed that the majority of the requests came from 72%-84% of the clusters in a DDoS attack, however, in case an FE, only 10% of the clusters contribute to the majority of the requests. It means that the distribution of clients among clusters is uniform in the case of a DDoS attack whereas it is highly skewed in case an FE. In the event of an FE, the number of requests for a particular requested file follows the Zipf-like distribution, whereas,



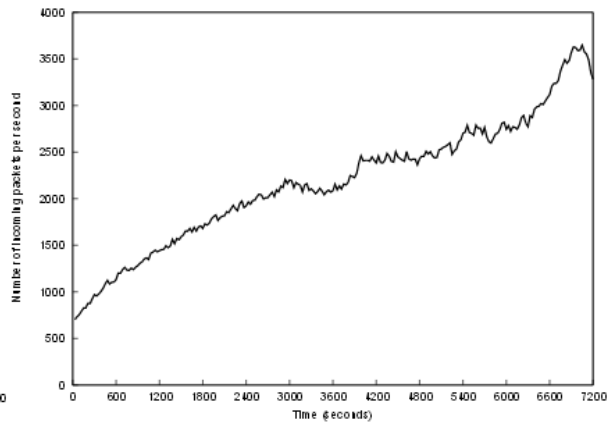
(a) Traffic Profile of CAIDA 2007 Attack Dataset



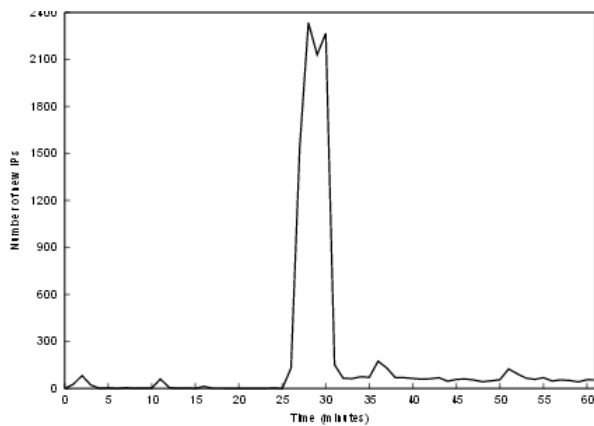
(b) Traffic Profile of 1998 FIFA World cup Dataset



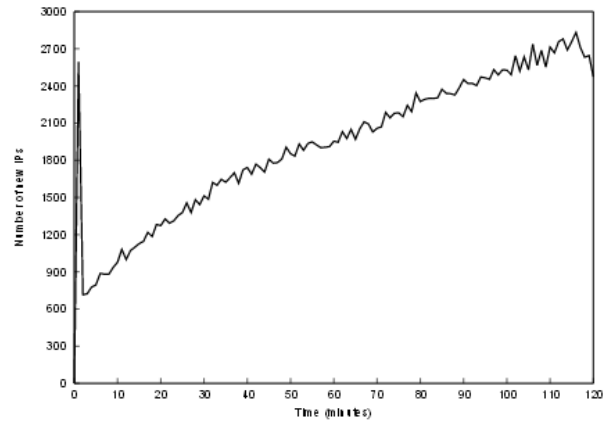
(c) Incoming traffic rate in CAIDA dataset



(d) Incoming traffic rate in 1998 FIFA World cup Dataset



(e) New Source IPs in CAIDA dataset



(f) New Source IPs in 1998 FIFA World cup Dataset

Figure 1: Incoming traffic profiles of CAIDA and FIFA world cup datasets

in a DDoS attack, the requests are more concentrated towards a set of files. However, their proposed parameters did not consider different scenarios of DDoS attacks. A sophisticated attacker can easily mimic the traffic pattern of the network to elude the detection system.

Bhatia et al. [5, 6] computed many parameters like the variation in the rate of new source IPs, change in incoming traffic rate, and the number of requests per source IP to exploit the behavioral difference between an FE and a DDoS attack. They validate their approach by using two real publicly available datasets of 1998 FIFA World Cup and CAIDA DDoS attack 2007. They observe that in a DDoS attack, a sudden burst of incoming traffic is experienced by the victim server within a short time as shown in Figure 1(c) whereas, in the case an FE, there is a gradual increase in incoming traffic to the web server over the time before hitting a maximum, as shown in Figure 1(d).

They found that the victim host sees a significant number of new IPs at the start of a DDoS attack and later, it sees very few new source IP addresses during an ongoing DDOS attack as shown in Figure 1(e). But regularly new source IP addresses are observed by the web server in the case of an FE as shown in Figure 1(f). Among the distinct source IPs, the distribution of traffic is more uniform in the event of a DDoS attack whereas, a small number of requests originate per source IP in case of an FE. They observed that the coefficient of variation is greater than 1 in an FE, which indicates Erlang distribution whereas it is less than 1 in a DDoS attack which signifies hyper-exponential distribution.

They also proposed a mathematical model for an FE with growing flash Phase and declining decay phase. They classify the flash events into predictable, unpredictable and secondary types. They use an information entropy metric to measure this component and observe that the resource entropy start to decrease with the outset of FE and remain to be same for the remaining flash phase. During decay phase, it begins to increase.

Sheng et al. [25] proposed a CALD system to protect the web server against DDoS attacks. They predict HTTP request rate & use a Kalman filter for calibrating the forecast results. They use the assumption that the rapid variation of traffic leads to the markable presence of abnormal traffic. They use entropy as the detection metric. They observe that the mess extent for DDoS is more as compared to FE. They validated their approach using real-time logs from two websites namely www.sina.com and www.taobao.com.

Saravanan et al. [22] proposed a behavioral detection system based on the rationale of flow similarity, client legitimacy and page referred to distinguish between FE and high rate application layer DDoS attacks. They use Hellinger Distance as the detection metric. They observed that the distance metric value is close to zero in the case of a DDoS attack, whereas it is close to one in case the of an FE as shown in Figure 2. They validate their proposed approach by simulating the CAIDA dataset & FIFA world-cup dataset for DDoS attack and FE respectively.

Their proposed work results in a small number of false positives and false negatives, with the detection accuracy of around 91%.

Thapngam et al. [23] proposed a behavior-based detection system. They use the rationale of packet arrival rate to discriminate DDoS attack from FE. They compute Pearson's correlation coefficient. They found that in the case of a DDoS attack, there is a high degree of automation along with predictable transmission rate, leading to correlation value close to 0 or 1. They observed that the request rate is unpredictable in the case of an FE which results in correlation value less than 1. For the evaluation of their proposed method, they use the data from 1998 FIFA world cup dataset and project mstream attack.

Prasad et al. [18] proposed three rationales namely the distribution of source IP addresses, access intent and speed of increased-decreased traffic to observe the traffic behavior of FE and DDoS attacks. They proposed an information theoretic Internet threat monitoring (ITM) system consisting of centralized data center and a group of monitors for the modeling and discrimination of FE attack from a DDoS attack.

Their proposed system work in two phases. In the first phase, they detect the ongoing attack by computing the entropy values and in the second phase, they discriminate the FE and DDoS attack using variation in entropy values. They deploy their proposed system on the Internet with the botnet.

Shui et al. [28] observed that the attack tools are usually similar for one botnet with the same pre-built program. In a single botnet, all the bots follow a single command by the bot master to start an attack session. In a botnet, the number of active bots is usually less than the number of legitimate users. So, to mimic a flash event, the active bots generate a significant number of packets which result in small standard deviation among attack flows than flash flows.

They found that the flow similarity in DDoS is much more than an FE. They propose a discrimination algorithm which computes flow correlation coefficient among suspicious flows. They validate their approach using a real dataset of 1998 FIFA World Cup for FE. A real DDoS attack tool called mstream is used to generate DDoS attack data.

Hakem et al. [3] proposed a connection-score scheme to detect application layer DDoS attacks. Their proposed system computes the statistical attributes such as download rate, request rate, uptime, downtime, classification of the page type, page popularity, hyperlink click rate, and hyperlink depth. They compute the values of these attributes to set the baseline behavior of the network, and scores are assigned to the various connections accordingly. The connections which get the lowest score are the malicious connections, and bottleneck resources are taken back from them by the server. They perform the experiment using Emulab using real traces of ClarkNet server. They observed that the Connection-Score scheme could tackle the application layer DDoS attacks efficiently as

Table 1: Comparison of related work for discriminating DDoS attacks and an FE

Sr. No.	Author/Year	Parameters	Detection Metric	Validation Technique	Datasets Used
1.	Saravanan et al. [22] / 2016	1. Flow Similarity 2. Page Referred 3. Client Legitimacy	Hellinger Distance	Simulation	1998 FIFA world-cup for FE CAIDA 2007 for DDoS attack
2.	Abhinav et.al[4] /2016	1. Page Access Order 2. Number of Source IPs 3. Flow Similarity 4. No. of Requests per IP 5. Unique Source IPs	Entropy	Simulation	1998 FIFA world-cup for FE CAIDA 2007 for DDoS attack
3.	Sachdeva et.al [21] /2014	1. Source IPs 2. Traffic Clusters	Entropy	Simulation Emulation	1998 FIFA world-cup for FE CAIDA 2007 for DDoS attack
4.	Prasad et.al [18] / 2013	1. Source IP Distribution 2. Access Intents 3. Change in Traffic Rate	Entropy	Realtime	-
5.	Tongguang et.al [15] /2013	1. HTTP-GET requests per Source IP	Entropy	Simulation	1998 FIFA world-cup for FE myDOOM botnet for DDoS attack
6.	Katiyar et.al[11] /2013	1. Source IP and port 2. Destination IP and port	Entropy	Simulation	-
7.	Yu et.al [28] /2012	1. Flow Similarity	Correlation coefficient	Simulation	1998 FIFA world-cup for FE mstream attack tool for DDoS attack
8.	Beitollahi et.al [3] /2012	1. Uptime and Downtime 2. Request rate and Download rate 3. Page popularity and Classification 4. Hyperlink depth and Click rate	Entropy	Emulation	Clarknet server logs
9.	Bhatia et.al [5] /2012	1. Volume of Incoming traffic 2. Number of Source IPs 3. Resource accessed	Entropy	Emulation	1998 FIFA world-cup for FE CAIDA 2007 for DDoS attack
10.	Thapngam et al. [23] /2011	1. Packet arrival rate	Correlation coefficient	Simulation	1998 FIFA world-cup for FE mstream attack tool for DDoS attack
11.	Wen et.al [25] /2010	1. Distribution of Source IPs 2. Page access order	Entropy	Realtime	NLANR Auckland VIII for FE www.sina.com for DDoS attack www.taobao.com for DDoS attack
12.	Yu et.al [27] /2009	1. Flow Similarity	Jeffrey Distance Sibson Distance Hellinger Distance	Simulation	NLANR Auckland VIII for FE MIT Lincoln for DDoS attack
13.	Li et.al[12] /2009	1. Source IPs Distribution 2. Access Intent 3. Traffic Rate	Total Variation Correlation coefficient		HTTP logs for FE MIT Lincoln for DDoS attack
14.	Oikonomou et.al[17] /2009	1. Access Content 2. Request Dynamics 3. Ability to ignore invisible content	Probability matrix	Simulation	Synthetically generated logs for FE Web server logs for DDoS
15.	Yatagai et.al [26] /2007	1. Page Access Order 2. Browsing Time 3. Page information	correlation metric	Realtime	-
16.	J.Jung et al. [10] / 2002	1. Traffic pattern 2. Cluster characteristics 3. File References	Entropy	Realtime	-

compared to existing methods.

Shui Yu et al. [27] proposed a detection algorithm for the discrimination of a DDoS and an FE. They compute information distance between different kinds of network flows, with the idea that the DDoS flows are strongly alike as compared to an FE because of similar pre-built programs executed by the attackers. They validate their approach by simulating the real datasets of NLANR PMA Auckland dataset for FE and MIT LLS DDOS 1.0 intrusion dataset for a DDoS attack.

They count the number of packets destined to a web server. Their proposed detection algorithm gives detection accuracy of 65% while discriminating DDoS and FE flows. They compute Sibson distance, Jeffrey distance, and the Hellinger distance and prove that the Sibson distance metric is better as compared to the other detection metrics for discriminating a DDoS attack from an FE.

Yatagai et al. [26] modeled an HTTP-GET flood detection technique by taking into account the page access behavior. They propose two detection algorithms. The first algorithm deals with the web page browsing order. If there are some IP addresses with the same browsing order, then, the GET requests from those IP addresses are dropped. Because it is assumed that in case a DDoS attack, all the participated attackers will generate an equal number of GET requests. The second algorithm computes the correlation between browsing time and page information size.

They found that in the case of regular clients, the browsing time increases in proportion to the information size. They deployed the proposed detection technique at the network gateway to computing the number of false positives and false negatives. They observed that when we give high priority to client services, then the first algorithm provides better results whereas, when the detection rate of HTTP-GET flood attack is given more priority, then the second algorithm is more appropriate.

Tongguang et al. [15] proposed a novel concept based on the entropy of HTTP-GET requests per source IP (HRPI) for the discrimination of AL-DDoS attack from legitimate traffic. They found that the HRPI value dramatically drops in case of a DDoS attack, however, in the event of an FE, there is an abnormal increase in HRPI of the network.

They proposed a two-step detection scheme. In the first step, the approximation of the Adaptive AutoRegressive(AAR) model to transform the HRPI time series to multidimensional vector series (MVS). Then, the support vector machine (SVM) is applied to classify AAR parameters to identify the attack. To validate the proposed approach, they simulate the MyDoom worm for application layer DDoS attack and use FIFA world-cup dataset for FE traffic. They observed that their approach could identify the DDoS attack traffic and FE with high precision, efficiency, and flexibility.

Li et al. [12] proposed a detection method using proba-

bility metrics for the discrimination of an FE and a DDoS attack. They compute a composite probability metric of total variation and similarity coefficient. The detection mechanism comprises a flow anomaly detector that identifies the specified router for observing the anomalies in incoming network flows. The flow distribution estimator estimates the distribution of sampled flows using pre-defined characteristics and calculates the total variation and similarity coefficient values in parallel of the two flows. The decision device makes the distinction between FE, DDoS, and a legitimate flow based on the value of detection metric and decides the type of anomaly.

To validate their approach, they use the legitimate and attack profile from the real dataset of MIT Lincoln Laboratory. For FE traffic, they use the HTTP log dataset from a busy server.

Oikonomou et al. [17] analyzed the human behavior for the discrimination of DDoS bots from human users. Their proposed detection scheme is based on three models. A request dynamic model for capturing the human's interaction with the server to detect bot aggressiveness. A request semantic model for capturing the common human request pattern to mark the bots that make different sequences. A deception model that model human invisible attributes into server replies. The addresses that request for these items are marked blacklisted. To validate their proposed approach, they use a collection of web server logs, and synthetically generated logs of FE bots. Their proposed approach gives detection accuracy of 95%.

Katiyar et al. [11] proposed a novel traceback mechanism based on entropy variations to discriminate a DDoS attack from an FE. They compute entropies of source IP, source port, destination IP and destination port. It is assumed that the router stores the entropy values of each flow during the non-attack period. Once a DDoS or an FE is identified, it starts to trace the source of the attack. They perform a simulation based experiment to validate their proposed approach. They evaluate traceback time, packet delivery ratio (PDR), and throughput based on entropy. They observed that the PDR under attack is small in comparison to an FE or a non-attack case. Throughput is maximum in case of non-attack case, nearly the same in an FE but it decreases in a DDoS attack.

Abhinav et al. [4] proposed a taxonomy of FEs based on the nature of events occurred, traffic generated, geographical distribution, the signature of the network, duration, and the shock level. They discriminate DDoS attacks from FEs by computing a set of packet header features like new source IPs, change in request rate, the number of distinct source IPs, page access entropy, the similarity between flows, and distribution of request rate among source IPs. They used FIFA World Cup 1998 dataset for FE, CAIDA dataset for simulating application-layer DDoS attack for validating their approach. They used curl-loader and Bonesi DDoS attack tool to simulate different scenarios of DDoS attacks.

Sachdeva et al. [19] used cluster entropy to discriminate FEs from DDoS attacks. Their detection approach is

based on the idea that during an FE, most of the requests comes from the already visited clients. They calculate the entropy of traffic clusters and source IPs. They observed that there is the significant increase in source IP entropy and minor variation in traffic cluster entropy in the case of an FE whereas in a DDoS attack, there is a substantial increase in source IP entropy as well as traffic cluster entropy. They perform emulation based experiments on DETER testbed, CAIDA dataset, and FIFA world-cup dataset to validate their approach.

4 Key Rationales to Discriminate DDoS Attacks from FEs

After the extensive review of existing research, we have derived a list of rationales that can be used to distinguish DDoS attacks from FEs as shown in Table 2.

5 Research Gaps

Today, the major thrust area in the field of DDoS attack detection is to distinguish the attack traffic from similar looking FEs. An FE occurs when a server experiences a sudden surge in the number of requests from legitimate clients. The FEs share many common characteristics with DDoS attacks such as a substantial increase in the incoming network traffic, the overloading of the servers providing the services, and degradation in the delivery of services. We have been able to find the following research gaps after the extensive review of existing research in discriminating DDoS attacks from FEs.

- Most of the researchers have validated their proposed approaches using publically available real datasets. They have mostly used 1998 FIFA world-cup dataset for FE traffic. This dataset seems to be obsolete if we consider the high-rate network traffic of nowadays fast growing networks but still, the pattern of GET requests towards a web server is still the same. However, the lack of availability of other related real datasets makes the validation a nightmare for the researchers.
- Most of the proposed solutions have used separate datasets for DDoS attacks and FE traffic. However, in reality, the DDoS attacks are often launched during FEs which makes the problem of discrimination very challenging. There are no real datasets available which contain the mixture of two types of traffic.
- Some researchers have tried to synthetically generate datasets using simulation and emulation based experiments [3, 4, 6, 17, 20] using a set of benchmark DDoS attack tools [2] but these datasets lack the capturing of relevant traffic features. Ideally, the captured network trace should contain the mixture of realistic background traffic and attack traffic in

Table 2: Summary of rationales to discriminate DDoS & FEs

Sr. No.	Rationales	DDoS	FE
1.	Flow Similarity	High	Low
2.	Web Pages Referred	Random	Hot pages only
3.	Client Legitimacy	unknown or new	Mostly well known
4.	Network Traffic Volume	Sharp increase and decrease	Gradual increase and decrease
5.	Change in Rate of New Source IPs	High in initial stage	High
6.	Distribution of clients	Uniform	Skewed
7.	Number of Distinct Clusters	Less and overlapped	Relatively more and new clusters
8.	Request-rate per source IP	More	Low
9.	Access Intents	To crash the server	Legitimate
10.	Distribution of source IPs	Limited with the availability of bots	Dispersive
11.	Correlation between browsing time & information size on web page	No effect	Increase
12.	Packet Delivery Ratio(PDR)	Low	High
13.	Throughput	Decreases	Maximum
14.	Web page browsing order	Same	Random
15.	Ratio of Entropy of source IPs and URL accessed	High	Low
16.	Duration of Traffic per Client	Long	Short
17.	Two way traffic	Low	High
18.	Coefficient of Variation	less than 1	greater than 1

appropriate proportion, and should not be biased towards a particular type of traffic. It is tough to ensure a proper mixture of normal and attack traffic in a real experiment driven dataset because there is no known formula to model Internet traffic correctly [9].

These research gaps clearly shows that it is very challenging to validate the proposed solutions to discriminate HR-DDoS attacks from FEs in the absence of latest publicly available real datasets. The availability of such realistic datasets that possesses the mixture of an appropriate attack traffic, non-attack traffic, and normal background traffic, is the need of the hour [1].

6 Conclusion

The detection of DDoS attacks is a challenging issue in the network security research. The problem is further magnified when such attacks are launched during a similar looking flash events (FEs). In this paper, we have comprehensively reviewed the prominent existing work done by the fellow researchers in the domain of discriminating DDoS attacks from FEs. We have also summarized a list of core rationales which have been used as detection metrics. This pragmatic list can further be extended and used for the future research in this domain to provide better practical solutions. As part of the future work, we shall propose an efficient detection and mitigation framework which would discriminate the DDoS attacks from FEs with a low false positive rate.

References

- [1] S. Behal and K. Kumar, "Trends in validation of DDoS research," *Procedia Computer Science*, vol. 85, pp. 7–15, 2016.
- [2] S. Behal and K. Kumar, "Characterization and comparison of DDoS attack tools and traffic generators- a review," *International Journal of Network Security*, vol. 19, pp. 383–393, May 2017.
- [3] H. Beitollahi and G. Deconinck, "Tackling application-layer DDoS attacks," *Procedia Computer Science*, vol. 10, pp. 432–441, 2012.
- [4] A. Bhandari, A. L. Sangal, and K. Kumar, "Characterizing flash events and distributed denial-of-service attacks: an empirical investigation," *Security and Communication Networks*, 2016.
- [5] S. Bhatia, G. Mohay, D. Schmidt, and A. Tickle, "Modelling web-server flash events," in *11th IEEE international symposium on Network computing and applications (NCA'12)*, pp. 79–86, 2012.
- [6] S. Bhatia, G. Mohay, A. Tickle, and E. Ahmed, "Parametric differences between a real-world distributed denial-of-service attack and a flash event," in *Sixth IEEE International Conference on Availability, Reliability and Security (ARES'11)*, pp. 210–217, 2011.
- [7] D. Braue, *Attack on Australian Census Site Didn't Register on Global DDoS Sensors*, Aug. 11, 2016. (<http://www.cso.com.au/article/604910/attack-australian-census-site>)
- [8] DDoS Attacks Net, *Recent DDoS Attacks*, Oct. 21, 2016. (<https://www.ddosattacks.net/twitter-amazon-other-top-websites-shut-in-cyber-attack/>)
- [9] S. Floyd and V. Paxson, "Difficulties in simulating the internet," *IEEE/ACM Transactions on Networking*, vol. 9, no. 4, pp. 392–403, 2001.
- [10] J. Jung, B. Krishnamurthy, and M. Rabinovich, "Flash crowds and denial of service attacks: Characterization and implications for cdns and web sites," in *Proceedings of the 11th International Conference on World Wide Web*, pp. 293–304, 2002.
- [11] P. Katiyar, U. Kumarn, and S. Balakrishanan, "Detection and discrimination of DDoS attacks from flash crowd using entropy variations," *International Journal of Engineering and Technology*, vol. 5, no. 4, pp. 3514–3519, 2013.
- [12] K. Li, W. Zhou, P. Li, J. Hai, and J. Liu, "Distinguishing DDoS attacks from flash crowds using probability metrics," in *Third International Conference on Network and System Security (NSS'09)*, pp. 9–17, 2009.
- [13] M. Mahmoud, M. Nir, and A. Matrawy, "A survey on botnet architectures, detection and defences," *International Journal of Network Security*, vol. 17, no. 3, pp. 264–281, 2015.
- [14] Arbor Networks, *DDoS Attack Report*, 2015.

- [15] T. Ni, X. Gu, H. Wang, and Y. Li, "Real-time detection of application-layer DDoS attack using time series analysis," *Journal of Control Science and Engineering*, vol. 2013, pp. 4, 2013.
- [16] K. Ohlson, *Victoria Secret Webcast of Their First Annual Online Show*, Feb. 5, 1999. (<http://edition.cnn.com/TECH/computing/9902/05/vicweb.idg/>)
- [17] G. Oikonomou and J. Mirkovic, "Modeling human behavior for defense against flash-crowd attacks," in *IEEE International Conference on Communications (ICC'09)*, pp. 1–6, 2009.
- [18] K. Prasad, A. Reddy, and K. Rao, "Discriminating DDoS attack traffic from flash crowds on internet threat monitors (ITM) using entropy variations," *African Journal of Computing & ICT*, vol. 6, no. 2, 2013.
- [19] M. Sachdeva and K. Kumar, "A traffic cluster entropy based approach to distinguish DDoS attacks from flash event using deter testbed," *ISRN Communications and Networking*, vol. 2014, 2014.
- [20] M. Sachdeva, K. Kumar, and G. Singh, "A comprehensive approach to discriminate DDoS attacks from flash events," *Journal of Information Security and Applications*, vol. 26, pp. 8–22, 2016.
- [21] M. Sachdeva, G. Singh, and K. Kumar, "An emulation based impact analysis of DDoS attacks on web services during flash events," in *2nd International Conference on Computer and Communication Technology (ICCCCT'11)*, pp. 479–484, 2011.
- [22] R. Saravanan, S. Shanmuganathan, and Y. Palanichamy, "Behavior based detection of application layer distributed denial of service attacks during flash events," *Turkish Journal of Electrical Engineering & Computer Sciences*, vol. 24, no. 12, pp. 510–523, 2016.
- [23] T. Thapngam, S. Yu, W. Zhou, and G. Beliakov, "Discriminating ddos attack traffic from flash crowd through packet arrival patterns," in *IEEE International Conference on Computer Communications Workshops*, pp. 952–957, 2011.
- [24] M. Uma and G. Padmavathi, "A survey on various cyber attacks and their classification.," *International Journal of Network Security*, vol. 15, no. 5, pp. 390–396, 2013.
- [25] S. Wen, W. Jia, W. Zhou, W. Zhou, and C. Xu, "Cald: Surviving various application-layer DDoS attacks that mimic flash crowd," in *4th International Conference on Network and System Security (NSS'10)*, pp. 247–254, 2010.
- [26] T. Yatagai, T. Isohara, and I. Sasase, "Detection of http-get flood attack based on analysis of page access behavior," in *IEEE Pacific Rim Conference on Communications, Computers and Signal Processing (PacRim'07)*, pp. 232–235, 2007.
- [27] S. Yu, T. Thapngam, J. Liu, S. Wei, and W. Zhou, "Discriminating DDoS flows from flash crowds using information distance," in *Proceedings of the third International Conference on Network and System Security (NSS'09)*, pp. 351–356, 2009.
- [28] S. Yu, W. Zhou, W. Jia, S. Guo, Y. Xiang, and F. Tang, "Discriminating DDoS attacks from flash crowds using flow correlation coefficient," *IEEE Transactions on Parallel and Distributed Systems*, vol. 23, no. 6, pp. 1073–1080, 2012.

Biography

Sunny Behal has done Bachelor of Technology in Computer Science and Engineering from SBS State Technical Campus, Ferozepur, Punjab, India in 2002. He finished his Masters in Computer Science and Engineering from Guru Nanak Dev Engineering College, Ludhiana, Punjab, India in 2010. Currently, He is full time Ph.D. Research Scholar at SBS State Technical Campus, Ferozepur. His research interests includes Botnet detection, DDoS attacks, Information and Network Security. He has published more than 40 Research papers in different International Journals and Conferences of repute.

Krishan Kumar has done Bachelor of Technology in Computer Science and Engineering from National Institute of Technology, Hamirpur in 1995. He finished his Masters in Software Systems from BITS Pilani in 2001. He finished his Ph. D. from Department of Electronics and Computer Engineering at Indian Institute of Technology, Roorkee in 2008. His general research interests are in the areas of Information Security and Computer Networks. He has published around 200+ research papers in different International Journals and Conferences of Repute including more than 500 citations.

Monika Sachdeva has done Bachelor of Technology in Computer Science and Engineering from National Institute of Technology, Jalandhar in 1997. She finished her Masters in Software Systems from BITS Pilani in 2002. She finished her Ph. D. from Department of Computer Science and Engineering at Guru Nanak Dev University, Amritsar, Punjab, India in 2012. Currently, she is working as Associate Professor in CSE Department at I.K.G. Punjab Technical University, Kapurthala, Punjab, India. His general research Interests are in the areas of Network Security and distributed computing. She has published more than 100 Research papers in different International Journals and Conferences.

Information Security Risk Management Framework for University Computing Environment

Umesh Kumar Singh¹, and Chanchala Joshi²

(Corresponding author: Chanchala Joshi)

School of Engineering and Technology, Vikram University Ujjain¹

Madhya Pradesh 456010, India

(Email: chanchala.joshi@gmail.com)

Institute of Computer Science, Vikram University Ujjain²

Madhya Pradesh 456010, India

(Received Aug. 22, 2016; revised and accepted Nov. 15 & Dec. 25, 2016)

Abstract

Today's universities are on the forefront of technological advancement which makes University' computing environment vulnerable because of its large open networks. This paper analyzed the security threats specifically evolve in University's network, and with consideration of these issues, proposed risk assessment framework for University computing environment. The proposed framework reduces the risk of security breach by supporting three phase activities; the first phase assesses the threats and vulnerabilities in order to identify the weak point in educational environment, the second phase focuses on the highest risk and create actionable remediation plan, the third phase of risk assessment model recognizes the vulnerability management compliance requirement in order to improve University's security position. The proposed framework is applied on Vikram University Ujjain India's, computing environment and the evaluation result showed the proposed framework enhances the security level of University campus network. This model can be used by risk analyst and security manager of University to perform reliable and repeatable risk analysis in realistic and affordable manner.

Keywords: Security Risk; Security Threats; University Campus Network; Vulnerability

1 Introduction

With increasing development of Information Technology, computing and network applications have become an integral part of universities environment. Today's universities are on the forefront of technological advancement. The greater access to technology results in valuable learning environment, on the other hand can also results vulner-

able computing environment with more security threats. University campuses are proving themselves to be some of the most technologically advanced places in the world by providing facilities like extensive Wi-Fi support, online learning using lecture capture software, digital library, classroom virtualization, web conferencing etc [23]. All these advancement makes University's computing environment particularly vulnerable because in contrast to hacking targets like banks, college and university computing environments are often large open networks. Protecting open large university campus against constantly evolving threats and vulnerabilities presents major challenges. On the other hand, the open computing university environment also supports diverse users; mainly the three distinct types of users of university are students, faculty and administration. Each of the user accesses university computing environment with varying level of university resources. Therefore, University campus network must not only provide the secure access to users but also defend them from vulnerabilities and security breaches. In the large University campus network there is need of improving risk posture and security effectiveness. It requires identification of operationally critical threats, assessment of vulnerabilities for measurement of risk level by continuous network monitoring of University campus network.

This paper proposes Quantitative Information Security Risk Assessment Model designed specifically for University computing environment, with the consideration of security dangers presents in large open campus network of University. The proposed model quantitatively measures the security risks by identifying potential threats and information processes within Universities network configuration. This model can be used by risk analyst and security manager of University to perform reliable and repeatable risk analysis in realistic and affordable manner.

2 Related Work

There are various risk assessment models available, some of which are qualitative while others are quantitative in nature; having a common goal of estimating the overall risk value [15]. OCTAVE (Operationally Critical Threat, Asset and Vulnerability Evaluation), developed by CERT is a model for risk-based infosec strategic assessment and planning [1]. OCTAVE defines assets as including people, hardware, software, information and systems. One of the major drawbacks of OCTAVATE is its complexity and it doesn't allow organizations to quantitatively model risk. In order to improve security organization system some standard principles are needed, Joshi et al. [8] analyzed the prominent taxonomies of attacks and vulnerability of computer system and network to improve vulnerability categorization and proposed novel approach towards Standardization of Network and Computer [7]. Harini et al. [5] proposed a simple, fast and efficient protocol for enhanced network architecture for authentication. One another prominent risk assessment model is [4] FAIR (Factor Analysis of Information Risk), provides framework for understanding, analyzing and measuring information risk. FAIR is built to address security concern weaknesses. The framework allows organizations to standardize the risk, apply risk assessment, view in total organizational risk, defend risk determination using advanced analysis and understand how time and money will affect the organization's security profile. The main shortcoming of FAIR is the lack of information about methodology and examples of how the methodology is applied. [6, 12] NIST RMF (National Institute of Standards and Technology's Risk Management Framework) covers a series of activities related to managing organizational risk. [21, 24, 25] TARA (Threat Agent Risk Assessment) is a risk assessment framework created by Intel that helps companies to manage risk by distilling the possible information about security attacks. The major drawback is to be prohibitively expensive and impractical to defend possible vulnerability. One of the primary tasks of risk assessment process is vulnerability scanning; Joshi et al. [9, 10] evaluated the efficiency of web application vulnerability scanners by designing a vulnerable web application. This evaluation assists in choosing vulnerability scanner during first phase of proposed model.

There are numerous risk assessment models; however, there is no mechanism to assist organizations in determining which model is the best to be employed within an organization; also these models considered the security challenges identified in hacking target organizations like banks. Although security risk assessment is crucial for these organizations but these organizations have secure and close network environment. On the other hand, higher educational institutions like Universities where information security risk assessment is major and high priority job are having large and open computing environment. The next section describes the typical scenario of University network environment comprises of diverse

small network.

3 University Campus Network Setup

Figure 1 shows an ideal, large and open, University campus network setup, comprises of diverse small networks. With the rapid development of technology, universities strive to develop a convenient and valuable learning environment through IT technologies. University large computing environment includes diverse network devices, various software applications and many servers. University network is large and open, so instead of trying to scan an entire network, we classify the hosts into groups and the scan each group.

- External Scan: Scanning through a router or firewall, 208.91.199.121.
- Internal Scan: The internal scan took place at the School of Engineering and Technology (SoET) location, and was plugged into a server that resides inside Vikram University's network.

In Figure 1 the placement of the blue scanner is inside the firewall, so it can scan internal vulnerabilities and the red scanner is used for external vulnerabilities scan. These internal and external vulnerability scans are used to collect data to assess the effectiveness of current security measures taken at the Vikram University's network. The internal scan took place at the School of Engineering and Technology (SoET) location, and was plugged into a server that resides inside Vikram University's network. The objective is to avoid external security counter measures to get a detailed view at system configurations. The external scan is for determining the security posture through Internet users view. The point behind external scanning is to identify what a hacker would see if he were trying to probe Vikram University's network.

4 Proposed Quantitative Information Security Risk Management Model

The main objective behind designing a security risk assessment framework is, "Security controls should be selected based on real risks to an organization's assets and operation". Numerous of security risks assessment models are available but University computing environment is differ from other organizations as it is large, open and consists of several small diverse network with various users. Selecting risk assessment model without analysis, results in implementation of security controls in the wrong places, wasting of resources and leaving an organization vulnerable to unanticipated threats. The proposed risk assessment model initially analyses what is to be assessed, who

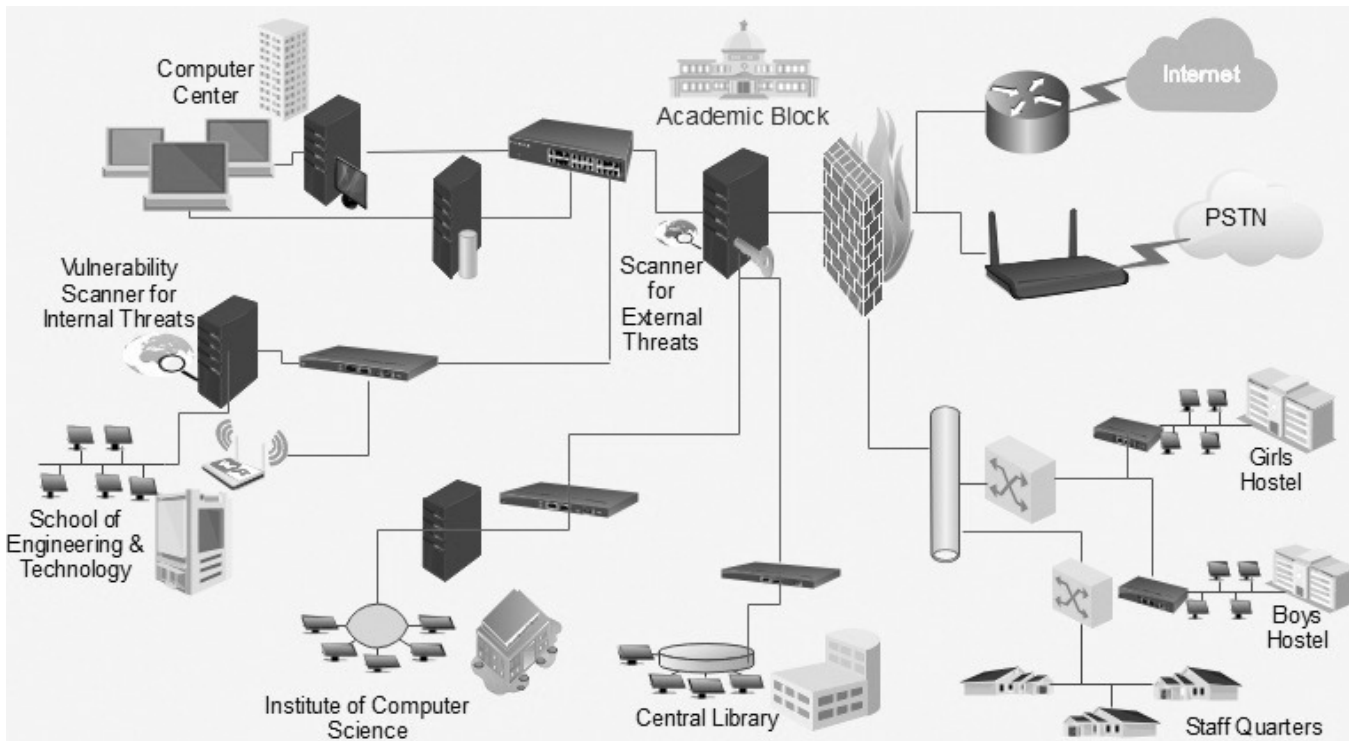


Figure 1: Network setup for Vikram university computing environment

needs to be involved and the criteria for quantifying, qualifying, and comparing severity of risks. The assessment results must be documented properly. The goal of proposed framework is to measure risk level quantitatively that will allow higher educational institutes to understand security risks. The proposed model is based on the most popular risk frameworks in use today, OCTAVE (Operationally Critical Threat, Asset, and Vulnerability Evaluation), developed at Carnegie Mellon University. The proposed framework performs three phase activities to make standard model more absolute, and provides a practical approach which can be used in real educational environment.

Figure 2 shows the abstract three phase view of the proposed model: The goal of proposed model is to reduce risks of security breach, this means understanding the cause that makes system vulnerable. The first phase focuses on knowing weak points, even in constantly changing and challenging University’s environment. Then the second phase concentrates on understanding which areas are having the highest risks, based on reliable and granular real risk scoring. The proposed framework uses Common Vulnerability Scoring System (CVSS) [13, 14] to validate which vulnerability can be actively exploited. The third phase pivot along the creation of actionable remediation plan over with University environment’s unique factor to and finally generate powerful reporting to track recursive risk measurement activities. The central of the proposed risk assessment framework is an objective of assessing University’s campus network, recursive mech-

anism that collects input regarding vulnerabilities and threats and produces quantitative risk level that can be measured and treated. General steps for the proposed framework are: identifying assets and stakeholders, understanding security requirements, assessing vulnerabilities, analyzing the effectiveness of controls, evaluation of risks by estimating frequency and impact of exploit, designing remediation plans and finally drive decisions using powerful reporting. Figure 3 shows the proposed framework for Quantitative Information Security Risk Assessment.

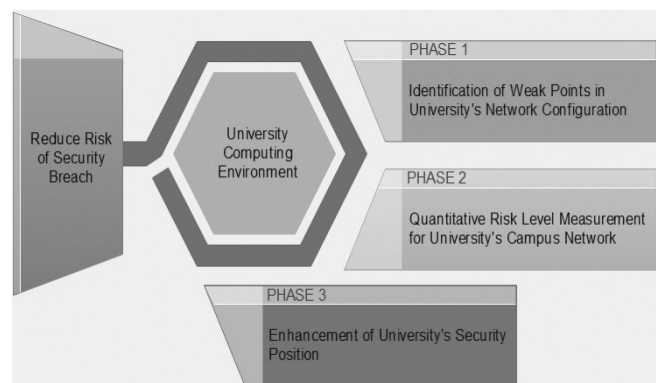


Figure 2: Three phases quantitative information security risk assessment model

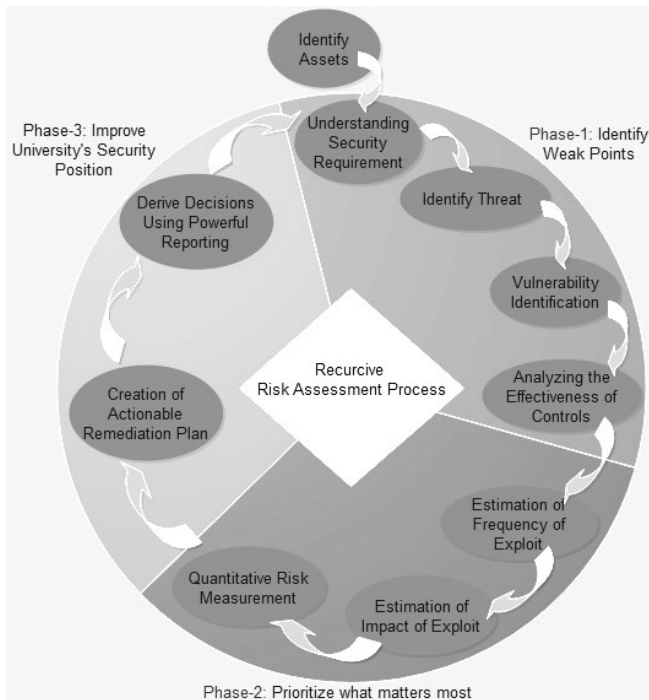


Figure 3: The proposed framework for quantitative information security risk assessment

4.1 Assets and Stakeholders Identification

The risk assessment techniques require to clearly specifying the assets. This step of proposed model defines the boundaries and contents of the asset to be assessed. In proposed framework information is taken as an asset.

4.2 Understanding Security Requirements

In this step, along with the resources and the information that constitute the system, the boundaries of the IT system will be identified. This step defines the scope of the risk assessment effort and provides information essential to defining the risk. The input for this step is information about hardware, software, data and information, network connections and system interfaces; and the output is a document that describes system mission, system boundary, system functions and information about criticality and sensitivity of data.

4.3 Threats and Vulnerabilities Identification

In this step, threat scenarios are created by listing the most common combinations of attack paths, attack goals and attack actor (attackers or hackers), that might lead to the compromise an asset.

4.4 Analysis of Effectiveness of Controls

In this step of assessment technical controls like authentication and authorization, intrusion detection, network filtering and routing, and encryption are considered and a document is prepared as an output which describes the effectiveness of system in defending against the particular threats.

4.5 Estimation of Frequency of Exploit

In this step, the likelihood that vulnerability can be exploited by the attacker is determined. Frequency of exploit will be calculated using mathematical formula and will be used in determining the quantitative security risk magnitude.

4.6 Estimation of Impact of Exploit

The impact can be measured by using Confidentiality Impact, Integrity Impact, and Availability Impact metrics of the CVSS [20]. The impact estimates how exploitation of a configuration issue could directly affect a targeted system and reflects the degree of loss of confidentiality, integrity, and availability. This step measures the impact of exploit onto the system.

4.7 Quantitative Risk Measurement

By the convergence of frequency and impact of exploit, quantitative security risk level can be measured. With the calculated risk magnitude the qualitative risk level can be determined in the range low to high. This risk level will be further used in creation of remediation plans.

4.8 Creation of Actionable Remediation Plan

Risk magnitude calculated in previous step prioritize the vulnerabilities which assists in defining remediation plans to validate identified vulnerabilities in order to improve system's security level. Second phase of the proposed identifies the areas are having the highest risks using Common Vulnerability Scoring System (CVSS) [20]. This risk magnitude can be used to estimate which vulnerability can be actively exploited and remediation plans will be designed using this information.

4.9 Drive Decisions Using Powerful Reporting

After completion of risk assessment procedure the results should be documented in an official report format. This report will help senior management, the mission owners in making decisions on policy, procedural, budget, and system operational and management changes. As risk assessment is recursive procedure, this final generated report will be used as an input of phase1 of proposed framework in the next cycle of risk assessment procedure.

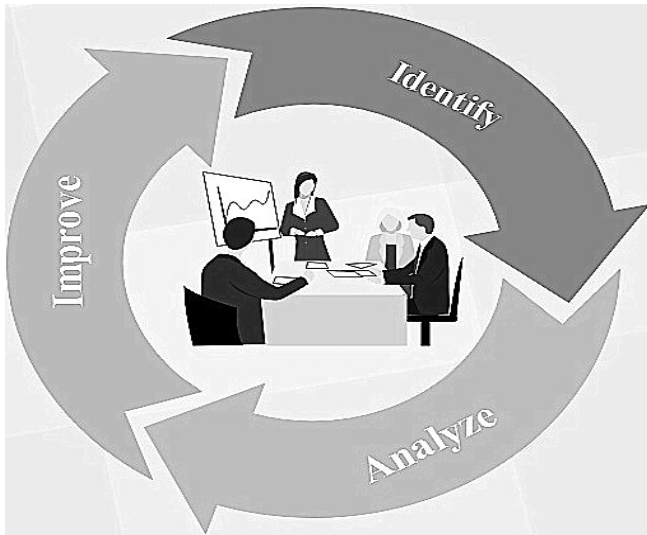


Figure 4: Ongoing risk assessment process

5 Evaluation of Proposed Quantitative Information Security Risk Assessment Model

As discussed in previous Section 2, University's network environment is continuously expanded and modified, its components changed, and its software applications replaced or updated with newer versions; these changes indicate that new risks will emerge and the previously mitigated risks may again become an issue. Thus, the risk management is ongoing and evolving process. This section emphasizes the good practice and need for an ongoing risk evaluation and assessment in order to improve security level. Figure 4 shows recursive model of ongoing risk assessment process.

In order to evaluate the importance and effectiveness of proposed model, it is applied on Vikram University Computing Environment (network setup of Vikram University shown in Figure 1).

5.1 Defining System Boundaries

The first phase of the proposed model identifies weaknesses and vulnerabilities visible and exploitable on the University computing environment. In the first step of first phase, the proposed approach for securing University campus network, determines information as an asset. The second step defines the scope of the effort, in risk assessment process. Characterizing the University's computing system establishes the scope of the risk assessment effort by identifying limits of the computing system along with resources and the information that constitute the network environment. The large and open network environment of Vikram University campus mainly suffers following security threats:

Phishing, ransomware, and malware.

Cybercriminals uses emails or Web accounts

that spoof official mailings for financial gain [18]. University's young students are at most of being the victim of a phishing attack that results in malware or ransomware downloads.

Wi-Fi. Vikram University provides Wi-Fi access on the University campus which is great in technology advancement view, but it can cause security problems in surprising ways.

Viruses Spreading through Social Media. Young adults of University are most avid users of social media like Facebook, Twitter and YouTube. This implies that in University's network malware can spread like wildfire through social media sites.

So Many Diverse Mobile Devices, so Much Risk.

Students are early adopters of technology, and new devices are frequently visible in campus; from iPads to new android phones, daily new launched devices are having upgraded versions of operating systems that can easily infected by smart attacker and also ready to infect University's network.

Embedded Devices are Risks Prone. Embedded connectivity improves the risks for viruses and more threats for network.

5.2 Vulnerability Identification and Assessment

After identification of the plausible security threats in university environment, the next step is to perform vulnerability assessment, which determines the potential impact of loss from a successful attack. Vulnerability scans apprised the administrator to the actual state of security on network and assist in defining remediation before an attacker discovers any vulnerability first. University network is large and open, so instead of trying to scan an entire network, we classify the hosts into groups and the scan each group. It will make scanning process easier. The scanning process is performed in two steps: external scan and internal scan. Since scanning through a router or firewall could hide internal vulnerabilities, therefore, as shown in Figure 1 of Vikram University network setup, the placement of the blue scanner is inside the firewall so it can scan internal vulnerabilities and the red scanner is used for external vulnerabilities scan. These internal and external vulnerability scans are used to collect data to assess the effectiveness of current security measures taken at the Vikram University's network. The internal scan took place at the School of Engineering and Technology (SoET) location, and was plugged into a server that resides inside Vikram University's network. The objective is to avoid external security counter measures to get a detailed view at system configurations. The external scan is for determining the security posture through Internet user's view. The point behind external scanning is to identify what a hacker would see if he were trying to probe Vikram University's network. The vulnerability scan requires the use

Table 1: Discovered hosts by Nexpose

Discovered	IP address	Host Name	OS	Services	Vulns
12/8/16 5:31 PM	208.91.199.121	vikramuniv.net	Unknown	465	72
12/8/16 5:31 PM	192.168.1.4	ICS	Windows Vista	7	23
13/8/16 12:44 AM	192.168.1.1	192.168.1.1	Linux	4	0

of scanning tools. The tools used to scan Vikram University were [16] Nexpose, [11] Metasploit and [9] Acunetix. The tool Nexpose is used to find hosts on the network have to be scanned for vulnerabilities. Acunetix is used for scanning web vulnerabilities while Metasploit is used along with Nexpose for penetration testing.

5.3 Major Findings

Nexpose placed within contact range of University's router, to find hosts and services on the network, discovered 35 hosts having 587 services, among which the main server of University is running with 27 high, 15 medium and 9 low vulnerability. Table 1 represents the format of result generated by Nexpose with some of the host's details.

Along with these details Nexpose generates details about active services, credentials and successful attacks. Details of vulnerabilities identified by Acunetix at host 208.91.199.121 are shown in Figure 5.

The snap shot of external scan results that summarized the identified alerts of the host 208.91.199.121 shown in Figure 6.

Alerts distribution

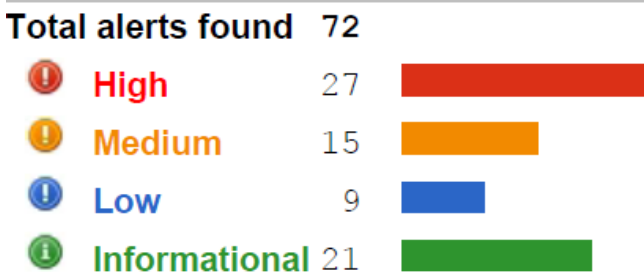


Figure 6: Web scan result of the host 208.91.199.121 by Acunetix web scanner1

Metasploit found 11 vulnerabilities during scan of the same host 208.91.199.121. Of these, 5 were critical, require immediate action because they are relatively easy for attacker to exploit and may provide them full control over the system. 5 vulnerabilities were severe, often harder to exploit and may not provide same access to affected systems. There was one moderate vulnerability discovered, provide information to attacker that may assist them in mounting subsequent attack in University network, so it should also be fixed in timely manner, but not urgent as other vulnerabilities.

Vulnerability scanners simply identify large numbers of

exposures and it is up to security teams to understand the severity of risks, which require knowledge of existing security infrastructure and additional manual effort. After identification of vulnerabilities present in Vikram University network environment, the next phase prioritized security vulnerabilities by calculating risk magnitude, along with estimation of frequency and impact of exploit.

5.4 Quantitative Risk Level Measurement

Risk assessment needed skilled individuals that understand probabilities, statistics and information technology. The first step in risk measurement requires integrating all scan results obtained from different scanners, Nexpose, Acunetix and Metasploit. Table 2 shows the scan results obtained by scanners Nexpose, Acunetix and Metasploit.

With all this vulnerability data gathered from scan results, security professional need to be able to prioritize risk by using as the Common Vulnerability Scoring System (CVSS) along with local network activity and device configurations. The risk level determines, among the identified vulnerabilities which of them actually create danger to the system and further, vulnerabilities are remediated according to the risk magnitude. Risk magnitude depends on the likelihood of the exploit, as the more frequent occurrences of vulnerability make system riskier; also, the Frequency of vulnerability depends on the date of emergence of vulnerability in the system [19]. The frequency and quantitative risk level of vulnerabilities determined by using the mathematical equations of Quantitative Security Risk Level Estimation Model [22], that computed temporal and environmental metrics to augment base CVSS scores and then derived a final risk value. The quantitative risk level score is ranging from 0 to 10; this numerical score can then be translated into a qualitative representation (such as low, medium, high, and critical) to help organizations properly assess and prioritize their vulnerability management processes. Table 3 interprets the risk rating values.

In quantitative risk level measurement along with severity of exploit, we are considering many factors like total number of alerts, affected item by exploit, affected parameter and variants identified during vulnerabilities scan. In Vikram University's computing environment the risk assessment method identified SQL injection, weak password and CSRF attacks at High risks.

Alerts (72)		Knowledge Base (4)		27	15	9	21	Generate Report
Start Date	19 Aug 2016 21:40	Files	15	Requests	17287	Host Name	http://vikramuniv.net	
End Date	19 Aug 2016 21:40	Directories	6	Avg. Response Time	142.78 ms	Scan Target Name	Vikram University Web Scan	
Duration	7h 56m 5s	Variations	14	Responsive	Yes	Scan Type	Web	
Name		Module						
+	●	Blind SQL Injection (6)		Scripting (Blind_Sql_Injection.script)				
+	●	Cross site scripting (verified) (1)		Scripting (XSS.script)				
+	●	Directory traversal (1)		Scripting (Directory_Traversal.script)				
+	●	Microsoft IIS tilde directory enumeration (1)		Scripting (IIS_Tilde_Dir_Enumeration.script)				
+	●	Script source code disclosure (1)		Scripting (Script_Source_Code_Disclosure.script)				
+	●	SQL injection (verified) (15)		Scripting (Sql_Injection.script)				
+	●	Weak password (2)		Scripting (Html_Authentication_Audit.script)				
+	●	Application error message (10)		Scripting (Error_Message.script)				
+	●	HTML form without CSRF protection (3)		Crawler				
+	●	User credentials are sent in clear text (2)		Crawler				
+	●	ASP.NET version disclosure (1)		Scripting (ASP_NET_Error_Message.script)				
+	●	Clickjacking: X-Frame-Options header missing (1)		Scripting (Clickjacking_X_Frame_Options.script)				
+	●	Cookie without HttpOnly flag set (1)		Crawler				
+	●	Cookie without Secure flag set (1)		Crawler				
+	●	Login page password-guessing attack (4)		Scripting (Html_Authentication_Audit.script)				

Figure 5: Acunetix Web scan result of the host 208.91.199.121

Table 2: Integrated scan results

Vulnerability	Severity	Total Alerts	Category
Weak password	7.5	2	A Brute Force attack
Weak password	7.5	2	Insufficient Authentication
Cross-site Scripting(verified)	4.4	1	Cross-site Scripting
Blind SQL Injection	7.8	6	SQL Injection
SQL injection (verified)	7.8	15	
Microsoft IIS tilde directory enumeration	2.6	1	Information Leakage
Script source code disclosure	2.6	1	
Weak password	7.5	2	
Application error message	5.0	10	
ASP.NET version disclosure	0.0	1	
Microsoft IIS version disclosure	0.0	1	
Password type input with auto-complete enabled	0.0	4	
Directory traversal	6.8	1	Path Traversal
HTML form without CSRF protection	8.6	6	Abuse of Functionality
Clickjacking: X-Frame-Options header missing	6.8	1	
Login page password-guessing attack	6.8	4	

Table 3: Qualitative risk rating scale

Quantitative Risk Magnitude	Risk Category	Description
9.0 to 10.0	Critical	Risk is totally unacceptable; must require immediate action to reduce likelihood of occurrence.
7.0 to 8.9	High	Risk is unacceptable; should require remediation plan to be implemented as soon as possible.
4.0 to 6.9	Medium	Risk may be acceptable over the short period of time; require that in future actions and budget plans to reduce risk should be included.
0.1 to 3.9	Low	Risks are acceptable; plans to further reduction of risk should be implemented with other security upgrades.

5.5 Recommendations for Up-gradations

Based on the findings from the risk assessment, the next phase of the proposed framework is to identify counter-measure upgrades that will reduce the risk levels. The previous phase of risk assessment identifies SQL injection, weak password and CSRF attacks at High risks in Vikram University's network. This section presents recommendations about identified risks in order to improve University network's security.

SQL injection. Vikram University computing environment identified total 21 SQL injection security alerts and the affected items are: /Login.asp, /Register.asp, /Search.asp, /showforum.asp and /showthread.asp. SQL injection attacks reshape the SQL queries which alter the nature of the program for the benefit of the hacker [2]. Server Side defense using Prepared Statement [9] is the most effective way to protect from SQL Injections, because it ensures that intent of query is not changed.

Weak Password. In University network 6 alerts of weak password are detected in /Login.asp. There are several accounts with passwords older than thirty days and some are even close to a year old. Password cracking is one of the most common elements used to assess the current security posture [3]. The simpler way to overcome weak password vulnerability is to enforce password policies, such as password length should be more than 8 characters, contains at least one capital and one small letter, at least one numeric and one special symbol should be included while choosing password.

CSRF Attacks. Total 6 variants detected with affected items /Login.asp, /Register.asp and /Search.asp. The CSRF vulnerabilities occur when applications allow every valid session identifier request to be processed by the application business logic [17]. The main threat is concerned to the way the browser handles requests. A simple example is a web application uses the GET method in an HTTP request for transferring password information; the browser encodes form data into a URL while using GET. Since form

data is in the URL, it is displayed in the browser's address bar, and information leakage occurs. The simplest solution is the use of POST method, while using the POST method, form data appears within the message body of the HTTP request, not the URL.

And finally, the risk assessment results are documented in an official report format which help senior management, the mission owners in making decisions on policy, procedural, budget, and system operational and management changes. As risk assessment is recursive procedure, this final generated report will be used as an input of phase1 of proposed framework in the next cycle of risk assessment procedure.

6 Conclusions

This paper proposed Quantitative Information Security Risk Assessment framework for University's Computing Environment. The goal of proposed model is to reduce risks of security breach, this means understanding the cause that makes University's campus network vulnerable. Applying the proposed framework onto the Vikram University campus network, it is clear that the current approaches of securing the network are ineffective in University environment's concern; as University's computing environment is differ in contrast to hacking targets like banks. The evaluation study addresses the issues found in Vikram University's network, such as enforcement of password policies, remote access management and restricting permissions to mandatory accounts. The proposed model quantitatively measured the risk magnitude for University's network configuration and can be used by risk analyst and security manager of University to perform reliable and repeatable risk analysis in realistic and affordable manner. The proposed framework can be applied to any higher educational organization or University's IT environments; it enables Universities to stay a step ahead of security threats and also to get more value from their security budget, by focusing on critical assets that are truly at risk.

Acknowledgments

The authors are thankful to MP Council of Science and Technology, Bhopal, for providing support and financial grant for the research work.

References

- [1] C. Alberts and A. Dorofee, *An Introduction to the Octave Method*, Technical Report PA 15213-3890, Software Engineering Institute, Carnegie Mellon University, Taiwan USA, Aug. 2003.
- [2] N. Asha, M. V. Kumar, and G. Vaidhyanathan, "Preventing sql injection attacks," *International Journal of Computer Applications*, vol. 52, pp. 28–32, Aug. 2012.
- [3] Y. Asimi, A. Amghar, A. Asimi, and Y. Sadqi, "Strong zero-knowledge authentication based on virtual passwords," *International Journal of Network Security*, vol. 18, pp. 601–616, July 2016.
- [4] B. Dixon, "Understanding the fair risk assessment," Nebraska CERT Conference, 2009. (<http://www.certconf.org/presentations/2009/files/TA-2.pdf>)
- [5] N. Harini and T. R. Padmanabhan, "3c-auth: A new scheme for enhancing security," *International Journal of Network Security*, vol. 18, pp. 143–150, Jan. 2016.
- [6] Joint Task Force Transformation Initiative, *Guide for Applying the Risk Management Framework to Federal Information Systems: A Security Life Cycle Approach*, Technical Report NIST Special Publication 800-37 Revision 1, Feb. 2010.
- [7] C. Joshi and U. K. Singh, "Admit - A five dimensional approach towards standardization of network and computer attack taxonomies," *International Journal of Computer Application*, vol. 100, pp. 30–36, Aug. 2014.
- [8] C. Joshi and U. K. Singh, "A review on taxonomies of attacks and vulnerability in computer and network system," *International Journal of Advanced Research in Computer Science and Software Engineering*, vol. 5, pp. 742–747, Jan. 2015.
- [9] C. Joshi and U. K. Singh, "Analysis of vulnerability scanners in quest of current information security landscape," *International Journal of Computer Application*, vol. 145, pp. 1–7, July 2016.
- [10] C. Joshi and U. K. Singh, "Performance evaluation of web application security scanners for more effective defense," *International Journal of Scientific and Research Publications*, vol. 6, pp. 660–667, June 2016.
- [11] D. Kennedy, J. O'Gorman, D. Kearns, and M. Aharoni, *Metasploit The Penetration Tester's Guide*, San Francisco: William Pollock, 2011.
- [12] Computer Security Division (Information Technology Laboratory), *Risk Management Framework (RMF)*, Technical Report SP 800-37 Rev. 1, Oct. 2016.
- [13] P. Mell, K. Scarfone, and S. Romanosky, "Common vulnerability scoring system," *IEEE Security and Privacy*, vol. 4, pp. 85–89, Nov.-Dec. 2006.
- [14] P. Mell, K. Scarfone, and S. Romanosky, "Cvss: A complete guide to the common vulnerability scoring system version 2.0," *Forum of Incident Response and Security Teams (FIRST)*, 2007.
- [15] 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.
- [16] W. Rosenberry, *Nexpose: Vulnerability Management and Penetration Testing System v.5.1 Security Target*, Technical Report 545 Boylston Street, Suite 400 Boston, MA 02116, May 2012.
- [17] K. Sentamilselvan, P. S. Lakshmana, and N. Ramkumar, "Cross site request forgery: Preventive measures," *International Journal of Computer Applications*, vol. 106, pp. 28–32, Nov. 2014.
- [18] J. Singh, "Cyber-attacks in cloud computing: A case study," *International Journal of Electronics and Information Engineering*, vol. 1, no. 2, pp. 78–87, 2014.
- [19] U. K. Singh and C. Joshi, "Quantifying security risk by critical network vulnerabilities assessment," *International Journal of Computer Applications*, vol. 156, no. 13, pp. 26–33, Dec. 2016.
- [20] U. K. Singh and C. Joshi, "Information security assessment by quantifying risk level of network vulnerabilities," *International Journal of Computer Application*, vol. 156, pp. 37–44, Dec. 2016.
- [21] U. K. Singh and C. Joshi, "Measurement of security dangers in university network," *International Journal of Computer Applications*, vol. 155, no. 1, pp. 6–10, Dec. 2016.
- [22] U. K. Singh and C. Joshi, "Quantitative security risk evaluation using cvss metrics by estimation of frequency and maturity of exploit," in *Proceedings of the World Congress on Engineering and Computer Science (WCECS2016)*, San Francisco, USA, Oct. 2016.
- [23] D. Stiawan, Y. Idris, H. Abdullah, and M. AlQurashi, "Penetration testing and mitigation of vulnerabilities windows server," *International Journal of Network Security*, vol. 18, pp. 501–513, May 2016.
- [24] T. T. Taiwhenua, *Risk Assessment Process*, Technical Report 3.0 NZ, Internal Affairs, NewZealand Government, Information Security, Feb. 2014.
- [25] J. Wynn, J. Whitmore, G. Upton, L. Spriggs, and D. McKinnon, *Threat Assessment and Remediation Analysis Tara*, Technical Report 031180SE-K1, Oct. 2011.

Biography

Umesh Kumar Singh received his Doctor of Philosophy (Ph.D.) in Computer Science from Devi Ahilya University, Indore(MP)-India. He is currently Associate

Professor in Computer Science and Director in School of Engineering and Technology, Vikram University, Ujjain(MP)-India. He has authored 6 books and his about 100 research papers are published in national and international journals of repute. He was awarded Young Scientist Award by M.P. council of Science and Technology, Bhopal in 1997. He is reviewer of various International Journals and member of various conference committees. His research interest includes Computer Networks, Network Security, Internet and Web Technology, Client-Server Computing and IT based education.

Chanchala Joshi received her Master of Science in Computer Science and Master of Philosophy in Computer Science from Vikram University, Ujjain(MP)-India. She is currently Junior Research Fellow and doctoral student in Institute of Computer Science, Vikram University, Ujjain(MP)-India. Her research interest includes network security, security measurement and risk analysis.

Speech Perceptual Hashing Authentication Algorithm Based on Spectral Subtraction and Energy to Entropy Ratio

Qiu-Yu Zhang¹, Wen-Jin Hu¹, Si-Bin Qiao¹, and Yi-Bo Huang²

(Corresponding author: Qiu-Yu Zhang)

School of Computer and Communication, Lanzhou University of Technology¹

No. 287, Lan-Gong-Ping Road, Lanzhou 730050, China

(Email: zhangqylz@163.com)

College of Physics and Electronic Engineering, Northwest Normal University²

No. 967, An-ning East Road, Lanzhou 730070, China

(Received Dec. 9, 2016; revised and accepted Mar. 1 & 12, 2017)

Abstract

In order to meet the requirements of robustness and discrimination of content preserving operations after conversion of speech communication format on the heterogeneous mobile terminal, and noise reduction and efficient authentication, a new efficient speech perceptual hashing authentication algorithm based on spectral subtraction and energy to entropy ratio was proposed. Firstly the proposed algorithm uses spectral subtraction method to denoise the speech signals which processed by applying pre-processing. Secondly, the energy to entropy value matrix of each frame is obtained by applying the method of energy to entropy ratio. Finally, the binary perceptual hash sequence is generated. Experiment results show that the proposed algorithm can denoise the speech effectively, and have good robustness and discrimination to content preserving operations, as well as having high efficiency and good ability to implement tamper detection.

Keywords: Energy to Entropy Ratio; Speech Noise Reduction; Speech Perceptual Hashing; Spectral Subtraction; Tamper Detection

1 Introduction

Currently, Android and iOS are the most popular mobile phone systems, code conversion is needed when there is a communication between two different systems, such as Android system and iOS system. Android's AMR (adaptive multi-rate) format should be converted to WAV format. So when one speech format is converted to another speech format, how to ensure the integrity and authenticity of the speech content? In addition, in the speech instant messaging, the speech is usually affected by coding and decoding, channel noise, delay, packet loss, and

the impact of the retrieval speed. In order to achieve efficient speech authentication, how to solve the problem of the interaction between robustness, distinguish and authentication efficiency, so it is very important to study the speech perceptual hashing authentication and speech noise reduction technology [1, 18, 19].

At present, the speech noise reduction methods mainly include: noise cancellation method, spectral subtraction, Wiener filtering method, Kalman filtering method, adaptive filtering method and so on. The spectral subtraction is one of the most commonly used methods. The speech perceptual hashing feature value extraction and processing methods mainly include: logarithmic cepstral coefficients [15], linear frequency spectrum [14], Mel-frequency cepstral coefficients [7, 16], linear prediction coefficient [12], Hilbert transform [22], space-time modulation [13], bark-bands energy [17] and so on. Huang *et al.* [7] proposed a speech perceptual hashing algorithm based on Mel-frequency cepstral coefficients (MFCC) combined with LPCC. The algorithm has good robustness and tamper localization, but it is not good at distinguishing and keeping the content of different speeches, in addition, the signal noise ratio is too high.

Chen *et al.* [4] proposed a speech perceptual hashing algorithm based on LPC combined with non-negative matrix factorization (NMF). The algorithm has good ability of collision resistance, but it is not effective to distinguish the different speeches and content preserving operations. Jiao *et al.* [9] proposed a LSF speech perceptual hashing algorithm based on compressed domain. The algorithm has good robustness and discrimination at low bit rate, but the LSF algorithm is of high computational complexity which affects real-time communication. Zhang *et al.* [20] proposed an efficient speech perception hashing algorithm based on a linear predictive residual coefficient

(LPR) of LP analysis combined with G.729 coding. The algorithm has good robustness, discrimination and high efficiency, but its robustness is poor when the signal noise ratio is low. Jiao *et al.* [8] proposed a speech perception hashing algorithm for the LSP parameterization of speech, which uses the discrete cosine transformation to extract the final characteristic parameters. The algorithm has a good compactness, randomness and collision resistance, but the extraction efficiency is not high.

Chen *et al.* [2] proposed a speech perception hashing algorithm, which conducts NMF operation on the matrix of the wavelet coefficients based on the wavelet transformation, and gets the hash value finally. Although the algorithm has good robustness in all kinds of content preserving operations, but its processing efficiency is low. Deng *et al.* [5] proposed a hashing algorithm which extracts perceptual feature value based on spectrum energy and divides the audio signal into 33 equal frequency sub-bands, and the energy of each sub-band is further processed by frequency time filter to get higher robustness to noise and channel distortion, each sub-band energy is represented by 2 bits to obtain the hash value after processing, but the performance is not good at low signal noise ratio (SNR). Huang *et al.* [6] proposed a speech perceptual hashing algorithm based on the improved LPC. The algorithm has good effect on the robustness and the sensitivity of the malicious attacks, and the authentication efficiency is high, but the effect is not very good in distinguishing and keeping the content of different speeches. Li *et al.* [10] proposed a speech perceptual hashing algorithm based on modified discrete cosine transform (MDCT) correlation coefficients combined with NMF. Although the algorithm has good robustness of content preserving operations, but the performance is poor in hashing extraction and matching authentication. Li *et al.* [11] proposed a speech perception hashing algorithm based on MFCC correlation coefficients combined with pseudo random sequences. The algorithm has good robustness, discrimination and security, but its collision resistance is poor and performance at the low signal noise ratio is not good. Chen *et al.* [3] proposed a speech perceptual hashing algorithm based on cochlea and cross recursion, which reduces dimensions by using NMF. The algorithm has good robustness, but the authentication efficiency is low.

In order to solve the problems above, we present an efficient perceptual hashing based on spectral subtraction and energy to entropy ratio for speech authentication after analyze the data that used spectral subtraction and without applying spectral subtraction. The proposed algorithm can solve the problem of the mutual influence between the robustness of content preserving operations, discrimination and authentication efficiency when the AMR format speech converted to WAV format. Firstly, preprocessing of the speech signal is performed after format conversion of the proposed algorithm. And then the spectral subtraction is used to denoise the speech signal. Secondly, the energy entropy ratio parameter matrix of each frame is calculated by using energy to entropy

ratio, and the final binary perceptual hashing sequence is generated. Finally, the hashing matching is performed by calculating the hashing number, and the integrity of the speech content is realized perfectly.

The rest of this paper is organized as follows. Section 2 describes the basic theory of spectral subtraction for noise reduction and energy to entropy ratio. A detailed Speech Perceptual Hashing Authentication scheme is described in Section 3. Subsequently, Section 4 gives the experimental results as compared with other related methods. Finally, we conclude our paper in Section 5.

2 Problem Statement and Preliminaries

2.1 Spectral Subtraction for Noise Reduction

The spectral subtraction is the most commonly used speech noise reduction method [21]. Let $s(n)$ be the time series of the speech signal, N represent the frame length, and $s_i(m)$ describe the i -th frame for speech signal after windowing and framing. Any frame of speech signal after performed discrete Fourier transform (DFT) is defined as in Equation (1):

$$S_i(k) = \sum_{m=0}^{N-1} s_i(m) \exp(j \frac{2\pi mk}{N}) \quad k = 0, 1, \dots, N-1. \quad (1)$$

Then the amplitude and phase angle of each component of $S(k)$ are obtained. The amplitude can be expressed as $|S_i(k)|$, and phase angle formula can be written as:

$$S_{angle}^i = \arctan \left[\frac{\text{Im}(S_i(k))}{\text{Re}(S_i(k))} \right]. \quad (2)$$

It is assumed that the length of time of no speech section which at the beginning of speech signal (noise clip) denoted as IS , and the corresponding frames are denoted as NIS . Then the average energy of the noise clip can be obtained:

$$D(k) = \frac{1}{NIS} \sum_{i=1}^{NIS} |S_i(k)|^2.$$

The calculation formula for spectral subtraction is shown as in Equation (3):

$$|\hat{S}_i(k)|^2 = \begin{cases} |\hat{S}_i(k)|^2 - a \times D(k) & |\hat{S}_i(k)|^2 \geq a \times D(k) \\ b \times D(k) & |\hat{S}_i(k)|^2 < a \times D(k) \end{cases} \quad (3)$$

where, a and b are two constants, a is defined as reduction factor and b is defined as gain compensation factor.

It can be inferred by Equation (3) that the amplitude is $|\hat{S}_i(k)|$ after performed by the method of spectral subtraction. Combining with Equation (2), the speech sequence $\hat{s}_i(m)$ that processed by the method of spectral subtraction can be obtained by the inverse fast Fourier

transform (IFFT). In this paper, we use the characteristic of the phase insensitive of the speech signal, and the phase angle information of original speech is directly used in the speech signal processed by the method of spectral subtraction.

2.2 Energy to Entropy Ratio

The core of the method of energy to entropy ratio is that the energy of speech section in the speech signal is upward bulge, and the spectral entropy value is less than the spectral entropy value of noise clip. The difference between the speech section and the noise section is more prominent by the method of the energy to entropy ratio. Supposing $s(n)$ is the time series of the speech signal, the i -th frame of speech signal denotes as $s_i(m)$ after processed by windowing and framing, and the length of frame denotes as N . Then energy of each frame is shown as follows.

$$E_i = \sum_{m=1}^N s_i^2(m). \quad (4)$$

On the basis of Equation (4), the calculation relationship of energy is improved as follows.

$$LE_i = \log_{10}(1 + E_i/c).$$

where, c is a constant. Because of the parameter c , when the parameter c is set to larger value and the amplitude of the energy E_i of each frame fiercely fluctuated and it will be decreased in the LE_i . So the noise and unvoiced section will be distinguished well by a optional parameter c . Parameter c is set to 2 in this paper.

Supposing speech signal in the time domain waveform denoted as $s(n)$, and the i -th frame of the speech signal which processed by applying windowing and framing denotes as $s_i(m)$. And then FFT is performed on $s_i(m)$ and the normalized spectral probability density function of each frequency component is defined as $p_i(k) = Y_i(k) / \sum_{k=0}^{N/2} Y_i(k)$. $Y_i(k)$ denotes the energy spectrum of the k -th line frequency component, $p_i(k)$ represents the probability density of the k -th frequency component of the i -th frame, and N is the length of the FFT. The short-time spectral entropy of each analysis speech frame is shown in Equation (5):

$$H_i = - \sum_{k=0}^{N/2} p_i(k) \log p_i(k). \quad (5)$$

Thus the energy to entropy ratio is denoted as $EEF_i = \sqrt{1 + |LE_i/H_i|}$.

3 The Proposed Scheme

The processing flow of the efficient perceptual hashing algorithm based on spectral subtraction and energy to entropy ratio for speech authentication is shown in Figure 1. The speech of Android's AMR format signal is converted

to WAV format by the server platform of client, when the Android system communicated with iOS system. Firstly, the pre-processing is needed to the speech signal. Secondly, the method of spectral subtraction is performed in order to denoise the speech. And then the speech is processed by applying windowing and framing. Finally, the method of energy to entropy ratio is used to obtain energy to entropy value.

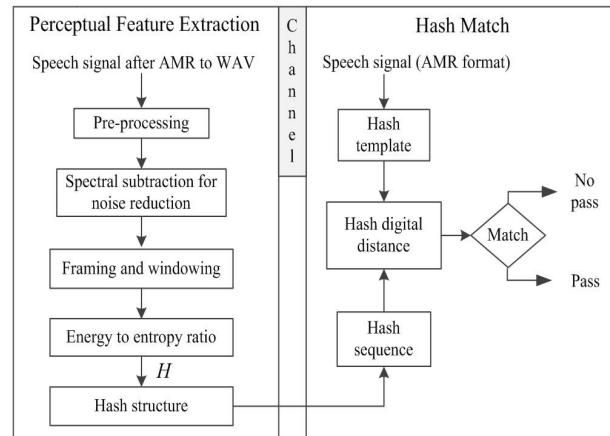


Figure 1: The flow chart of proposed algorithm

The hashing structure and matching of the speech signal are performed, and the processing steps are as follows:

Step 1: Pre-processing. The speech signal $s'(n)$ is obtained by pre-emphasis processing for the input signal $s(n)$. It is useful to improve the high frequency useful part of the signal and extract the subsequent feature. The sampling frequency of the speech signal $s(n)$ is 16 kHz, the number of channels is single channel, and the sampling precision is 16 bit.

Step 2: Spectral subtraction for noise reduction. The speech signal $s'(n)$ is processed by spectral subtraction, and then the speech signal $s''(n)$ is obtained. In the spectral subtraction experiment, the parameters are set as below: the length of frame is 30 ms, frame shift is 25 ms, $NIS=8$, $a=3$ and $b=0.5$. Different selection of the experimental parameters has significant impact on the results (especially noise). The above parameters are the optimal value after testing the experiment.

Step 3: Framing and windowing. The smoothed frame edge is added for speech signal $s''(n)$ by Hamming window. The length of frame is m . It is supposed that the speech $s''(n)$ is divided into n frame, and signal $A_i = \{A_i(k) | i = 1, 2, \dots, n, k = 1, 2, \dots, m\}$ is obtained.

Step 4: Energy to entropy ratio. Firstly, FFT is performed on each frame signal A_i , then the frequency domain signal $F_i = \{F_i(k) | i = 1, 2, \dots, n, k = 1, 2, \dots, m\}$ is obtained. Secondly, the energy value

of signal F_i is calculated through logarithmic energy algorithm, and then the spectral entropy value of signal F_i is calculated by spectral entropy algorithm. Finally, use the energy to entropy ratio to obtained the parameter matrix $\mathbf{G}(1, n)$, the parameter matrix $\mathbf{G}(1, n)$ is obtained by using the method of energy to entropy ratio, the middle value of matrix $\mathbf{G}(1, n)$ is extracted and it is added in the last new line of the matrix. The matrix $\mathbf{G}(1, n)$ is transformed into matrix $\mathbf{H}(1, n + 1)$.

Step 5: Hashing construction. Binary hashing construction is performed by \mathbf{H} , the hashing sequence \mathbf{h} is obtained, and the perceptual hashing sequence of speech signal $s(n)$ is $\mathbf{h}(1, n)=[\mathbf{h}]$.

The binary hashing construction method is as follows.

Using the parameter matrix in the first row of data to subtract the next line of data, if the result is more than 0, the line data turn into 1, otherwise 0.

$$h(i) = \begin{cases} 1 & H(i) > H(i+1) \\ 0 & H(i) \leq H(i+1) \end{cases} \quad i = 1, 2, \dots, n.$$

Step 6: Hash digital distance and matching. The bit error rate (BER) is defined as normalized hamming distance $D(:, :)$ of the perceptual hashing sequence that is derived from two speech clips s_1 and s_2 , namely, the ratio of the error bit number to the total number of the perceptual hashing value. The calculation formula is shown as follows:

$$D = \frac{\sum_{i=1}^N (|h_{s1} - h_{s2}|)}{N} = \frac{\sum_{i=1}^N (h_{s1} \oplus h_{s2})}{N}.$$

where, D is the BER, h_{s1} and h_{s2} correspond to the perceptual hashing values generated by speech clip s_1 and s_2 , and N is the length of the perceptual hashing values.

The probability of the appearance of “0” and “1” sequence is equal in theory, and the average normalized hamming distance is $0.5N$. We use the hypothesis test of the BER to describe the hashing matching.

P_0 : Two speech clips s_1 and s_2 are the same clip if $D \leq \tau$.

P_1 : Two speech clips s_1 and s_2 are different clip if $D > \tau$.

The hashing values of the same speech clips will take some changes if it be processed by content preserving operations. By setting the size of matching threshold τ , the perceptual hashing sequence mathematical distance of the speech clips s_1 and s_2 are compared. If the two mathematical distances $D \leq \tau$, and their perceptual content are treated as the same, the certification is passed, otherwise it doesn't pass the certification.

4 Experimental Results and Analysis

The speech data used in the experiment is the voice in the Texas Instruments and Massachusetts Institute of Technology (TIMIT) and the Text to Speech (TTS) speech library, which is composed of different contents recorded both in Chinese and English by men and women. Every speech clip is converted to WAV format by AMR format with the same length 4 s, which is of the form of 16 bits PCM, mono sampled at 16 kHz, the bit rate is 256 kbit/s, and the length of frame is 30 ms. The speech library in this paper is a total of 1,280 speech clips consisting of 600 English speech clips and 680 Chinese speech clips. The operating experimental hardware platform is Intel(R) Core(TM) i5-2410M CPU, 2.30 GHz, with computer memories of 4G. The operating software environment is MATLAB R2013a of Windows 7 system.

4.1 Robustness Test and Analysis

The content preserving operations are performed for the 1,280 speech clips, as shown in Table 1. The comparison results in various BER and running time between the proposed algorithm and the algorithm without applying spectral subtraction method are shown in Table 2.

As can be seen from Table 2, the proposed algorithm has good robustness and higher operating efficiency for increasing and decreasing of the volume, filtering, resampling and re-encoding than that without applying spectral subtraction algorithm. This is due to the above content preserving operations have little effect on energy and spectral entropy of speech section, at the same time, the algorithm is simple, so it has good robustness and efficiency. However, the noise has great influence on the method of spectral entropy, so the effect is not good on the speech added noise whether it is 20 dB or 30 dB. But the echo is relatively significant influence on the speech section energy, the mean is still high. We can analyze the data from Table 2, when applying spectral subtraction method, we can see that the mean values of all content preserving operation are decrease, but the running efficiency is improved by nearly one times. It has a good improvement on the volume adjustment, echo, resampling and Gaussian noise, this is because of the above operations have great influence on the speech amplitude and noise clip, so the effect is improved obviously by applying spectral subtraction method. Filtering and re-coding has little influence on no speech section which at the beginning of speech signal (noise clip) and the speech amplitude, so the effect of improvement is not remarkable. However, the spectral subtraction method increased the computational complexity and decreased the efficiency. The speech signal to noise ratio is obtained after the speeches processed by spectral subtraction method: the average SNR of 20 dB speech increased by 6.1993 dB and the average SNR of 30 dB speech increased by 6.2538 dB.

Table 1: Content preserving operations

Operating means	Operation method	Abbreviation
Volume Adjustment 1	Volume down 50%	V.↓
Volume Adjustment 2	Volume up 50%	V.↑
FIR Filter	12 order FIR low-pass filtering, Cutoff frequency of 3.4 kHz	F.I.R
Butterworth Filter	12 order Butterworth low-pass filtering, Cutoff frequency of 3.4 kHz	B.W
Resampling 1	Sampling frequency decreased to 8 kHz, and then increased to 16 kHz	R.8→16
Resampling 2	Sampling frequency increased to 32 kHz, and then dropped to 16 kHz	R.32→16
Echo Addition	Echo attenuation 25%, delay 300 ms	E.A
Narrowband Noise 1	SNR=30 dB narrowband Gaussian noise, center frequency distribution in 0 ~ 4 kHz	G.N1
Narrowband Noise 2	SNR=20 dB narrowband Gaussian noise, center frequency distribution in 0 ~ 4 kHz	G.N2
MP3 Compression 1	Re-encoded as MP3, and then decoding recovery, the rate is 32 kbit/s	M.32
MP3 Compression 2	Re-encoded as MP3, and then decoding recovery, the rate is 192 kbit/s	M.192

Table 2: The comparison results in various BER and running time

Algorithm	Spectral subtraction algorithm					Without applying spectral subtraction algorithm				
	Mean	Variance	Max	Time (s)	Average time (s)	Mean	Variance	Max	Time (s)	Average time (s)
V.↓	0.0007	0.0023	0.0149	117	121.2	0.0119	0.0112	0.0597	65	64.5
V.↑	0.0175	0.0198	0.0896	121		0.0291	0.0270	0.1343	62	
F.I.R	0.0504	0.0196	0.1269	123		0.0529	0.0246	0.1493	64	
B.W	0.0369	0.0172	0.1194	126		0.0359	0.0207	0.1343	63	
R.8→16	0.0081	0.0084	0.0448	121		0.0119	0.0119	0.0672	65	
R.32→16	0.0004	0.0018	0.0149	116		0.0006	0.0022	0.0149	64	
E.A	0.1042	0.0287	0.2090	122		0.1185	0.0308	0.2239	60	
G.N1	0.0770	0.0314	0.2164	128		0.0990	0.0518	0.3433	64	
G.N2	0.1363	0.0360	0.2836	124		0.1684	0.0568	0.3806	64	
M.32	0.0208	0.0145	0.0821	118		0.0249	0.0201	0.1119	70	
M.192	0.0027	0.0047	0.0299	117		0.0039	0.0061	0.0299	68	

The results of comparison between the proposed algorithm and the algorithm of Ref. [4], the average BER are shown in Table 3.

Table 3: Comparison of average BER

Operating means	Proposed	Ref. [4]
V.↓	0.0007	0.0726
V.↑	0.0175	0.1123
F.I.R	0.0504	0.3428
B.W	0.0369	0.3445
R.8→16	0.0081	0.1004
R.32→16	0.0004	0.0163
E.A	0.1042	0.1886
G.N1	0.0770	0.4615
M.32	0.0208	0.1682
M.192	0.0027	0.1009

As shown in Table 3, the average BER of the proposed algorithm underwent above attacks is lower than the algorithm of Ref. [4], which shows that our algorithm has good robustness on the content preserving operation, especially on volume controlling, resampling and re-coding. And it is also far superior to the algorithm in Ref. [4] about the 30 dB Gaussian noise and filtering.

This paper totally get 816,003 BER values by conducted pairwise comparison between perceptual hash val-

ues from 1,280 different speech clips, and the false accept rate (FAR) and false reject rate (FRR) is obtained via above attacks, and drawing the FAR-FRR curve, the results of comparison between without applying spectral subtraction method and the algorithm in Ref. [4] are shown in Figure 2.

The above FAR-FRR curve is without the content preserving operation of 20 dB Gaussian noise. As shown in Figure 2(a), the FAR-FRR curve obtained by the proposed algorithm is not cross, which means that the proposed algorithm has good distinction and robustness, and it can identify the content of the content preserving operation and the different speech content accurately. As shown in Figure 2(b), when did not apply spectral subtraction method, the FAR-FRR curve of the algorithm was cross, this is due to the poor effect in the Gaussian noise, and the problem of discrimination and robustness cannot be solved very well. As shown in Figure 2(c), the FAR-FRR curve obtained by the algorithm in Ref. [4] is cross, and the problem of discrimination and robustness cannot be solved very well. Combined with Table 2 and Table 3, we can conclude that the robustness on the content preserving operations of the proposed algorithm is better than the algorithm in Ref. [4] and the algorithm without applying spectral subtraction method. Moreover, the noise greatly reduced after applying the spectral subtraction method and the balance with discrimination, robustness

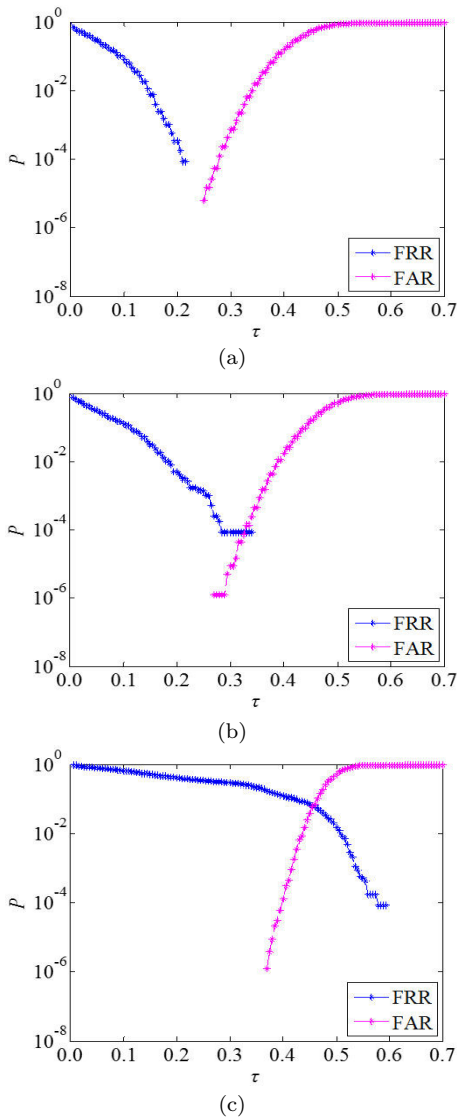


Figure 2: FAR-FRR curve of different algorithms. (a) The proposed algorithm, (b) The without applying spectral subtraction method, (c) The algorithm of Ref. [4].

and efficiency can be solved very well.

4.2 Discrimination Test and Analysis

The BER of the perceptual hashing values of different speech contents basically obeys the normal distribution. By pairwise comparison of perceptual hash values for 1,280 speech clips, there are 816,003 BER values are obtained. The normal distribution of the BER values of the perceptual hashing sequences is shown in Figure 3.

According to the central limit theorem of De Moivre-Laplace, the hamming distance approximately obeys normal distribution. When adopting BER as the distance measure, the BERs approximately obey a normal distribution ($\mu = p, \sigma = \sqrt{p(1-p)/N}$), where N is the length of perceptual hashing sequence. The closer the BER distribution curve is to the normal distribution, the better

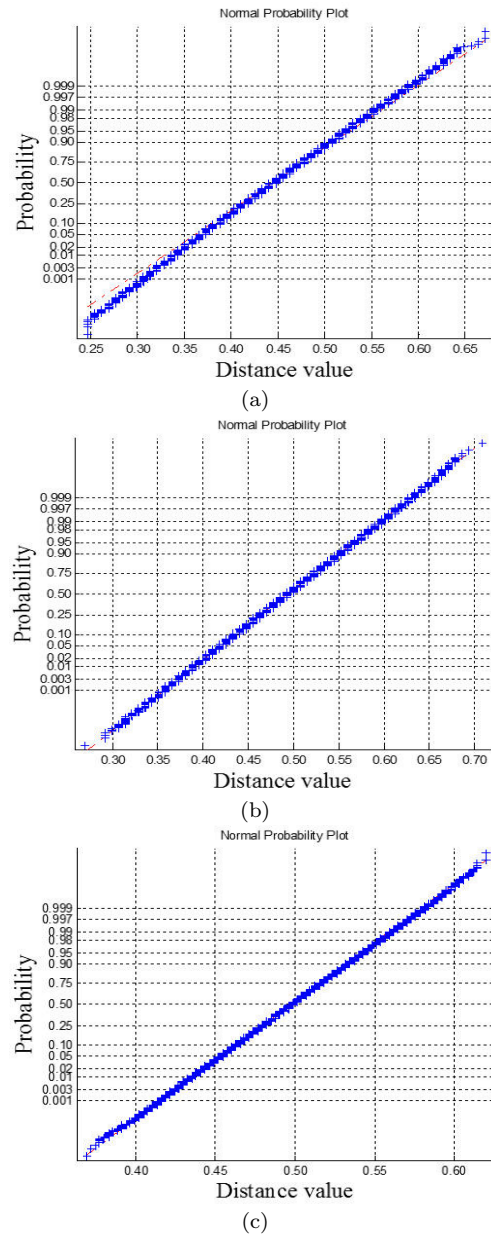


Figure 3: BER normal distribution diagram. (a) The proposed algorithm, (b) The without applying spectral subtraction method, (c) The algorithm of Ref. [4].

the randomness and collision resistance of the perceptual hashing sequence. In this paper, the length of perceptual hashing sequence is $N=134$. The theoretical normal distribution parameters mean and standard deviation $\mu=0.5, \sigma=0.0307$ that are obtained according to the central limit theorem of De Moivre-Laplace. The experimental results demonstrate that the mean and standard deviation are $\mu_0=0.4452, \sigma_0=0.0463$ in the proposed scheme. However, if without applying the spectral subtraction method in the proposed algorithm, the corresponding mean value is $\mu_1=0.4933$, and the standard deviation is $\sigma_1=0.0446$. The FAR is calculated in order to verify the correctness of the

experiment. The expression is shown as follows:

$$FAR(\tau) = \int_{-\infty}^{\tau} f(x|\mu, \sigma)dx = \int_{-\infty}^{\tau} \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}} dx.$$

where, τ is perceptual authentication threshold, μ represents the BER mean, σ is called BER variance, x is called false acceptance rate.

The comparison results of FAR value are shown in Table 4.

As shown in Table 4, the smaller the matching threshold τ is, the smaller the FAR value is. When the matching threshold $\tau=0.23$, there are approximately 1.67 speech clips misjudged in 1×10^6 speech clips, it demonstrates that the algorithm could meet the requirement of perceptual hashing authentication. By comparison with the algorithm of without applying spectral subtraction methods it can be obtained that the FAR is far lower than the algorithm that applying spectral subtraction method. It is because that when applying spectral subtraction some speech clips are regarded as noise therefore the distinction is decreased. So it is necessary to improve the spectral entropy method to reduce the FAR. When the algorithm can distinguish between the different speeches and the content preserving operations completely, the $\tau=0.2$ and FAR is 1.111×10^{-5} in Ref. [10], the $\tau=0.3$ and the FAR is 9.731×10^{-5} in Ref. [11], so the FAR of the proposed algorithm is lower than the Ref. [10, 11].

4.3 Tamper Detection and Localization

The speech instant messaging of mobile terminals are vulnerable to malicious tampering and attack of criminals. In order to achieve safe and reliable speech content authentication, the speech perception hash algorithm needs to possess the function of tamper detection and location ability for preventing illegal malicious attack and tampering. Generally, illegal malicious operation will cut or tamper part of the speech, errors under the content preserving operations of the speech are often distributed uniformly. However, errors caused by illegal malicious operation usually cause a greater impact in part of the area. So we can determine whether it has been tampered by comparing the hash value. Since the algorithm adopted in this paper is the binary perceptual hash value. So we can judge if there exist tampering by comparing perceptual hash value.

Calculated according to the standard speed 220 words per minute, if there are two or greater speech frames perceptual hashing values are different, we can affirm that it is tampered. This is because that the generally speaking speed is much faster than the standard. And it is also judged as tampering part in the case of the previous and latter frame is different, and the middle frame is same. Because when computing the perceptual hashing values, the previous frame hash value is affected by the latter frame hash value. In order to verify the sensitivity of the algorithm to malicious attacks or tamper, in the experiment, we select a clip of 4 s speech randomly;

different speech from the same speaker is used to replace 10% speech clips. Figure 4 is the schematic diagram of perceptual hashing value of tamper localization, where the red elliptic curves contain regions that are tampered. It can be known that the algorithm has a certain ability of tamper detection, and has a good accuracy of tamper detection and localization.

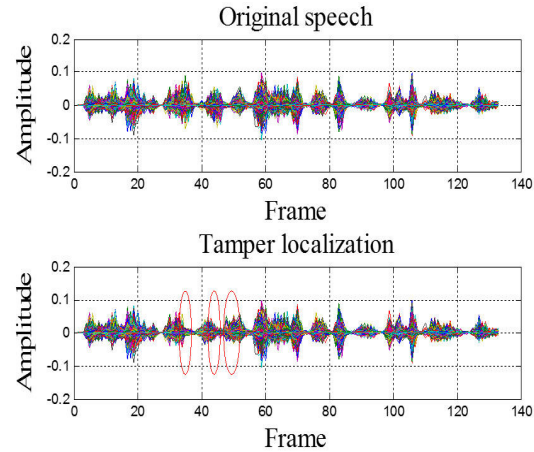


Figure 4: Tamper localization schematic diagrams

4.4 Efficiency Analysis

In order to assess the computational complexity and efficiency of the proposed algorithm, the average run-time is required when performing 1,280 speech clips which are selected randomly from the speech library. The comparison results of the proposed algorithm with the algorithm in Ref. [3, 4, 10] are show in Table 5. In Table 5, the file lengths are 4 s.

Table 5: Comparison of operating efficiency of the algorithm

Algorithms	Basic frequency (GHz)	Average running time (s)
proposed	2.30	0.0503
Without applying spectral subtraction	2.30	0.0299
Ref. [3]	3.30	0.9008
Ref. [4]	2.27	0.1603
Ref. [10]	2.50	0.1304

As shown in Table 5, the proposed algorithm efficiency is three times more faster than the Ref. [4], two times more faster than the Ref. [10], and nearly 18 times faster than the Ref. [3]. The proposed algorithm has high efficiency and low complexity, and the size of perceptual hashing sequence is 134, which is almost 1/15 of ($N = 64 \times 8 \times 4$) the algorithm in Ref. [8]. And the size of perceptual hashing sequence in the algorithm of Ref. [4, 10] is 360, which

Table 4: The comparison results of FAR value

τ	Proposed	Without applying spectral subtraction	Ref. [10]	Ref. [11]
0.10	4.4688×10^{-14}	5.8061×10^{-19}	2.939×10^{-12}	2.976×10^{-15}
0.15	9.0999×10^{-11}	6.9481×10^{-15}	1.144×10^{-8}	2.631×10^{-12}
0.20	5.9217×10^{-8}	2.4126×10^{-11}	1.111×10^{-5}	9.687×10^{-9}
0.23	1.6763×10^{-6}	1.7784×10^{-9}	-	-
0.25	1.2435×10^{-5}	2.4465×10^{-8}	2.715×10^{-4}	1.484×10^{-6}
0.30	8.5614×10^{-4}	7.3185×10^{-6}	1.682×10^{-3}	9.731×10^{-5}

shown that the summary of the proposed algorithm is powerful. Therefore, the proposed algorithm can meet the requirements of real-time and low complexity of speech communication, which can be applied to the mobile devices with limited bandwidth speech communication terminal and lower hardware configuration in mobile computing environment.

5 Conclusions

An efficient speech perceptual hashing authentication algorithm is proposed based on the spectral subtraction and energy to entropy ratio. The algorithm uses the spectral subtraction method to denoise the speech signal, and then the energy to entropy value that obtained by the method of energy to entropy rate as the perceptual feature which is used to construct the hash sequence and the speech is authenticated. Finally the robustness, discrimination and efficiency of the applied spectral subtraction method and without applying spectral subtraction method are analyzed. Simulations show that the robustness (especially noise) of the proposed algorithm is superior to that without applying spectral subtraction method, but the efficiency is reduced by nearly 1 times and the FAR is increased. In the different speech content preserving operations, the proposed algorithm can effectively resist on the conventional operations, such as resampling, echo, filtering, etc. Especially the effect is good at the volume adjustment and resampling. The proposed algorithm can fully distinguish the different speeches and content preserving operations, at the same time, the false accept rate is low, the efficiency is high, the summary of the proposed algorithm is powerful, and it has a good accuracy of tamper detection and localization.

The main disadvantage of the proposed algorithm is that the efficiency is reduced and the FAR is increased after applying the spectral subtraction method. The next of the research objective is to improve the spectral subtraction in order to decrease the impact of Gaussian noise and reduce the FAR of the algorithm, as well as achieve the approximate recovery and encryption of the speech tampering.

Acknowledgments

This study is supported by the National Natural Science Foundation of China under grant NSFC 61363078, the Natural Science Foundation of Gansu Province of China (No. 1606RJYA274). The authors gratefully acknowledge the anonymous reviewers for their valuable comments.

References

- [1] J. Chen, S. Xiang, H. Huang, and W. Liu, "Detecting and locating digital audio forgeries based on singularity analysis with wavelet packet," *Multimedia Tools and Applications*, vol. 75, no. 4, pp. 2303–2325, 2016.
- [2] N. Chen, H. D. Xiao, and W. G. Wan, "Audio hash function based on non-negative matrix factorisation of mel-frequency cepstral coefficients," *IET Information Security*, vol. 5, no. 1, pp. 19–25, 2011.
- [3] N. Chen, H. D. Xiao, J. Zhu, J. J. Lin, Y. Wang, and W. H. Yuan, "Robust audio hashing scheme based on cochleagram and cross recurrence analysis," *Electronics Letters*, vol. 49, no. 1, pp. 7–8, 2013.
- [4] N. Chen and W. G. Wan, "Robust speech hash function," *ETRI journal*, vol. 32, no. 2, pp. 345–347, 2010.
- [5] J. Deng, W. Wan, R. Swaminathan, X. Yu, and X. Pan, "An audio fingerprinting system based on spectral energy structure," in *Proceedings of the IET International Conference on Smart and Sustainable City (ICSSC'11)*, pp. 1–4, Shanghai, China, July 2014.
- [6] Y. B. Huang, Q. Y. Zhang, and Z. T. Yuan, "Perceptual speech hashing authentication algorithm based on linear prediction analysis," *TELKOMNIKA Indonesian Journal of Electrical Engineering*, vol. 12, no. 4, pp. 3214–3223, 2014.
- [7] Y. B. Huang, Q. Y. Zhang, Z. T. Yuan, and Z. P. Yang, "The hash algorithm of speech perception based on the integration of adaptive MFCC and LPCC," *Journal of Huazhong University of Science and Technology (Natural Science Edition) (in Chinese)*, vol. 43, no. 2, pp. 124–128, 2015.
- [8] Y. H. Jiao, L. Ji, and X. M. Niu, "Robust speech hashing for content authentication," *IEEE Signal Processing Letters*, vol. 16, no. 9, pp. 818–821, 2009.
- [9] Y. H. Jiao, Q. Li, and X. M. Niu, "Compressed domain perceptual hashing for MELP coded speech," in *Proceedings of the IEEE International Conference on*

- Intelligent Information Hiding and Multimedia Signal Processing (IIHMSP'08)*, pp. 410–413, Haerbin, China, Aug. 2008.
- [10] J. F. Li, H. X. Wang, and Y. Jing, “Audio Perceptual Hashing Based on NMF and MDCT Coefficients,” *Chinese Journal of Electronics*, vol. 24, no. 3, pp. 579–588, 2015.
- [11] J. F. Li, T. Wu, and H. X. Wang, “Perceptual Hashing Based on Correlation Coefficient of MFCC for Speech Authentication,” *Journal of Beijing University of Posts and Telecommunications (in Chinese)*, vol. 38, no. 2, pp. 89–93, 2015.
- [12] P. Lotia and D. M. R. Khan, “Significance of Complementary Spectral Features for Speaker Recognition,” *International Journal of Research in Computer and Communication Technology*, vol. 2, no. 8, pp. 579–588, 2013.
- [13] X. Lu, S. Matsuda, M. Unoki, and S. Nakamura, “Temporal modulation normalization for robust speech feature extraction and recognition,” *Multimedia Tools and Applications*, vol. 52, no. 1, pp. 187–199, 2009.
- [14] M. Nouri, N. Farhangian, Z. Zeinolabedini, and M. Safarina, “Conceptual authentication speech hashing base upon hypotrochoid graph,” in *Proceedings of the 6th IEEE International Conference on Symposium Telecommunications (IST'12)*, pp. 1136–1141, Glance, Iran, Nov. 2012.
- [15] H. Özer, B. Sankur, N. Memon, and E. Anarim, “Perceptual audio hashing functions,” *EURASIP Journal on Applied Signal Processing*, vol. 2005, no. 12, pp. 1780–1793, 2005.
- [16] V. Panagiotou and N. Mitianoudis, “PCA summarization for audio song identification using Gaussian Mixture models,” in *Proceedings of the 18th IEEE International Conference on Digital Signal Processing (DSP'13)*, pp. 1–6, Santorini, Greece, July 2013.
- [17] M. Ramona and G. Peeters, “Audio identification based on spectral modeling of bark-bands energy and synchronization through onset detection,” in *Proceedings of the 2011 IEEE Int. Conference on Acoustics Speech and Signal Processing (ICASSP'11)*, pp. 477–480, Prague, Czech, May 2011.
- [18] S. J. Xiang and J. W. Huang, “Audio watermarking to D/A and A/D conversions,” *International Journal of Network Security*, vol. 3, no. 3, pp. 230–238, 2006.
- [19] B. Q. Xu, Q. Xiao, Z. X. Qian, and C. Qin, “Unequal protection mechanism for digital speech transmission based on turbo codes,” *International Journal of Network Security*, vol. 17, no. 1, pp. 85–93, 2015.
- [20] Q. Y. Zhang, Z. P. Yang, Y. B. Huang, S. Yu, and Z. W. Ren, “Robust speech perceptual hashing algorithm based on linear predication residual of G.729 speech codec,” *International Journal of Innovative Computing, Information and Control*, vol. 11, no. 6, pp. 2159–2175, 2015.
- [21] Y. Zhang and Y. Zhao, “Real and imaginary modulation spectral subtraction for speech enhancement,” *Speech Communication*, vol. 55, no. 4, pp. 509–522, 2013.
- [22] H. Zhao, H. Liu, K. Zhao, and Y. Yang, “Robust speech feature extraction using the hilbert transform spectrum estimation method,” *International Journal of Digital Content Technology and its Applications*, vol. 5, no. 12, pp. 85–95, 2011.

Biography

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

Wen-jin Hu graduated from Shenyang Ligong University, Liaoning, China, in 2010. He received M.Sc. degrees in Communication and information system from Lanzhou University of Technology, Lanzhou, China, in 2014. His research interests include audio signal processing and application, multimedia authentication techniques.

Si-bin Qiao received the BS degrees in communication engineering from Lanzhou University of Technology, Gansu, China, in 2015. His research interests include audio signal processing and application, multimedia authentication techniques.

Yi-bo Huang received Ph. D. candidate degree from Lanzhou University of Technology in 2015, and now working as a lecturer in the College of Physics and Electronic Engineering in Northwest Normal University. He main research interests include Multimedia information processing, Information security, Speech recognition.

An Unsupervised Method for Detection of XSS Attack

Swaswati Goswami¹, Nazrul Hoque¹, Dhruba K. Bhattacharyya¹, Jugal Kalita²
(Corresponding Author: Dhruba K. Bhattacharyya)

Department of Computer Science and Engineering & Tezpur University¹
Napaam, Sonitpur, Assam-784028, India
(Email: dkb@tezu.ernet.in)

Department of Computer Science, University of Colorado²
Colorado Springs, O 80933-7150, USA²

(Received Jan. 3, 2016; revised and accepted May 20 & Apr. 9, 2016)

Abstract

Cross-site scripting (XSS) is a code injection attack that allows an attacker to execute malicious script in another user's browser. Once the attacker gains control over the Website vulnerable to XSS attack, it can perform actions like cookie-stealing, malware-spreading, session-hijacking and malicious redirection. Malicious JavaScripts are the most conventional ways of performing XSS attacks. Although several approaches have been proposed, XSS is still a live problem since it is very easy to implement, but difficult to detect. In this paper, we propose an effective approach for XSS attack detection. Our method focuses on balancing the load between client and the server. Our method performs an initial checking in the client side for vulnerability using divergence measure. If the suspicion level exceeds beyond a threshold value, then the request is discarded. Otherwise, it is forwarded to the proxy for further processing. In our approach we introduce an attribute clustering method supported by rank aggregation technique to detect confounded JavaScripts. The approach is validated using real life data.

Keywords: Attribute Clustering; Divergence; Malicious Script; Proxy; XSS

1 Introduction

Cross-site Scripting (XSS) is one of the most common application layer hacking techniques. It allows an attacker to embed malicious JavaScript, VBScript, ActiveX, HTML or Flash into a vulnerable dynamic page to fool the user, executing the script on his/her machine in order to gather data [17]. Most common way of stealing cookies or hijacking session is to embed a JavaScript encoded with browser supported HTML encoding technique. XSS attacks are categorized into three types [10]: reflected XSS, stored XSS and Document Object Model or DOM-based XSS

attack.

As the Internet applications are becoming more and more dynamic, the possibilities of such attacks have become more prominent. The number of vectors which are used to carry out such attacks are increasing with the increase in interactivensness of an application. Severeness of XSS attack can easily be predicted as it is ranked in top positions in recent security related surveys. For example, XSS is ranked third in the "OWASP Top 10 Application Security Risks-2013" [32].

Most of the existing intrusion detection systems which are designed to detect the XSS attack consider that, XSS attack is substantially caused by the failure of a Web application to check the contents for malicious codes before running it in the user's browser. The existing approaches can be categorized into three basic types [27]: dynamic approach, static approach, and hybrid approach. Static analysis includes various methods such as taint propagation analysis [20], string analysis [31], software testing techniques [28], etc. Taint propagation analysis includes construction of a control flow graph, where each node contains a label. An Web page is considered vulnerable, if the input node of the control flow graph for a certain variable has an edge leading to the output node. In string analysis, the program generated string values contain formal language expressions, such as Context Free Grammar (CFG) with labels. Minamide's method [24] of string tainting, which is an example of string analysis approximates string output of a program with a CFG. Software based testing techniques such as fault injection, penetration testing are used to infer the existence of vulnerabilities. Dynamic analysis includes proxy based solutions [21], browser enforced embedded policies [19], etc. In proxy based solutions, requests from the client side are intercepted in the proxy and based on the rules of the proxy the required actions are taken. On the other hand, in browser enforced embedded policies client is provided with a list of benign scripts by the Web application

and only these scripts are run. Although the static and dynamic solutions are effective in various cases, but in some situations the combination of both the approaches is much needed. Hence, the hybrid approaches are introduced. Sanar [3] is such a tool which combines static and dynamic approaches. In static analysis, it models the data input methods to indicate sanitization process. On the other hand, the code irresponsible for sanitization is reconstructed by dynamic analysis approach.

Machine learning based approaches use statically and dynamically extracted characteristic features from both malicious and benign samples and build classification tools [4].

1.1 Motivation

Although several methods have been introduced so far to mitigate XSS attack, it is still a live problem. The attack instances are increasing continuously and intruders are introducing more complex ways for embedding their scripts to trick users. Motivation behind choosing JavaScripts is that, now a days most of the web applications use JavaScripts extensively and the XSS attacks reported so far are in maximum cases found to be executed using JavaScripts. Moreover, already existing incremental approaches show a high false alarm rate and are not scalable [5]. Taking the whole scenario into consideration we are motivated to introduce a faster, stable and cost effective detection mechanism which will ensure high detection accuracy. Our aim is to reduce the false alarm rate and to increase the scalability.

1.2 Contributions

The two major contributions of this work are:

- A load balanced Client-Server based architecture to support XSS attack detection.
- An attribute clustering technique to support feature-level unsupervised grouping of attack and normal scripts over relevant and optimal feature space.

The remainder of this paper is organized as follows. The background of XSS attacks and a brief discussion on why we have concentrated mainly on reflected XSS attack is discussed in Section 2. Section 3 gives an overview of related works. Section 4 introduces our proposed method which is followed by experimental results in Section 5. Finally, we conclude our paper in Section 6.

2 Background and Related Work

In this section, we discuss the basics of XSS attacks, their categories and characteristics.

2.1 Basics of XSS Attacks

XSS attack mainly occurs due to the improper sanitization and validation of the user inputs given in the form of scripts. Figure 1 depicts a typical scenario of XSS attack. It shows, how an attacker can easily generate an XSS attack by sending a mail to the user containing a malicious URL. In the first step, attacker crafts a URL containing the malicious script and e-mail it to the victim. In Step 2, user clicks on the link send by the attacker and on clicking the link, the script is sent to the web server as the user request as shown in Step 3. In Step 4, the server reflects back the request to the user and the script is executed in the user's browser. Once the script is executed, sensitive data like session cookies are sent to the attacker in Step 6. Then attacker gets control over the user's session and can access the Web server on behalf of the user. The methods through which one can execute an XSS attack can be categorized into the following three types.

2.1.1 Persistent or Stored XSS Attack

Persistent or Stored XSS attack is server database related and it can affect a numerous number of users visiting the server which contains the malicious script in its database injected by the attacker. This type of attack mainly occurs due to the improper validation of the user inputs. Let us take an example to clarify the statement. A guest-book, which is a visitors log through which they can post their query or just leave a comment or give feedback for some services provided by a Website can be an easy victim of persistent XSS attack. Suppose a malicious user crafts a special script for cookie stealing and posted that as a comment in the guest-book. This malicious link may be a link to provoke the user for getting free recharge by posting a link with the tagline "*Hey check this link. I got free Recharge!!!*". If the server is not able to sanitize the input properly then this comment is saved to the server database. Now the visitors visiting that particular Webpage will execute that javascript in their browser unknowingly. The attacker will thus get the cookies of the user's browser and thus will get the control over the user's session.

2.1.2 Non-Persistent or Reflected XSS Attack

Recently, non-persistent or reflected XSS attacks have been found as a common type of XSS attack. Here, victim's request itself contains the malicious string. The server then responds with an HTML page that contains the script and thus the script is executed in the user's browser. Let us consider the following scenarios.

Scenario 1: Email is one of the most common ways of tricking a user to click on a malicious script. The attacker can send a link to the user via an email crafting a link which contains the malicious link. As and when the link is clicked by the user, the script which is hidden either in the link itself or in a script

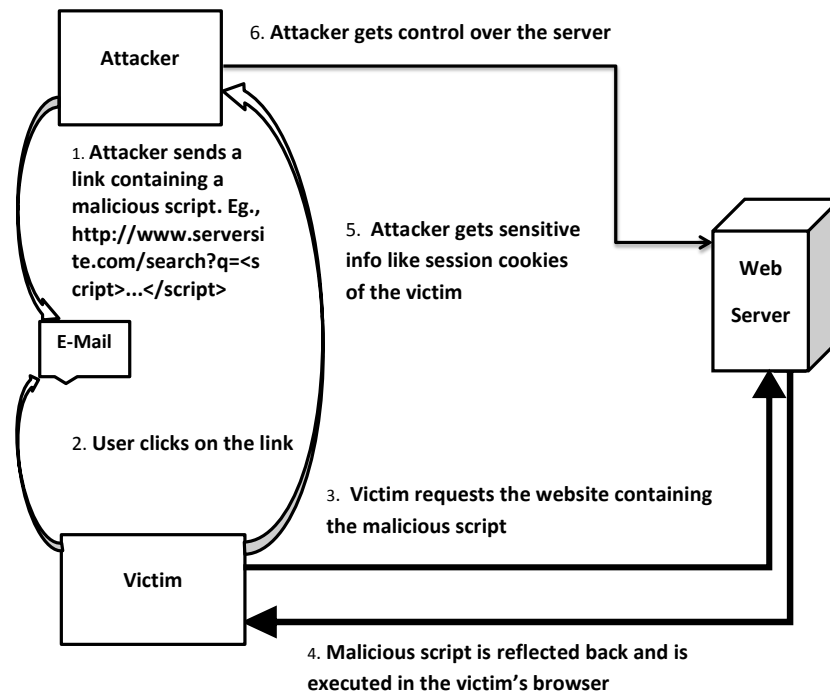


Figure 1: An overview of XSS attack

which the attacker refers to is executed and users credentials such as session cookies. are sent to the attacker. The attacker can easily get access to the site which the legitimate user is surfing. Figure 2(a) shows the scenario of this attack.

Scenario 2: We can consider yet another scenario of XSS attack. Here, the attacker acts as an intermediary agent. The attacker can be the host of a legitimate Website. When the user visits the attacker's Website, then the attacker prompts with a specially crafted link. When the user clicks on the link, it redirects the user to another Website to which the user have access to. This reflected message can contain a script, which is then executed in the user's browser and the attacker can get the browser info this way. The link may contain a page which actually doesn't exist on the requested server. Then the server sends back a message to the user saying that *page not found*. This scenario is depicted in Figure 2(b).

2.1.3 DOM-based XSS Attack

DOM-based XSS attack is the type of XSS attack that occurs in the Document Object Model (DOM) of an HTML page in lieu of the part of an HTML page. Here, since the changes occur to the DOM environment, so the HTTP response code runs in a different manner. DOM XSS attack can be carried out with a numerous DOM objects as mentioned below.

- User name or password part of a location or URL:

Here the payload is received by the server in the authentication header.

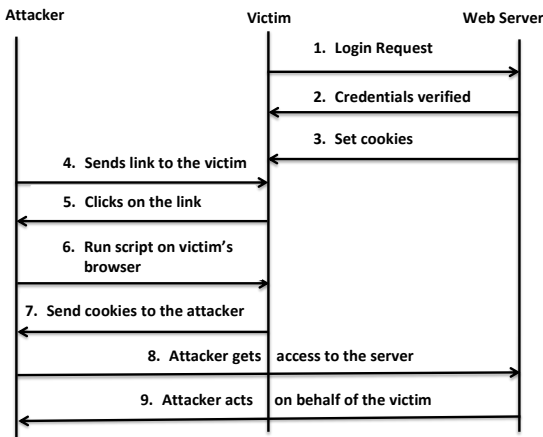
- Portion where the query part is located in the URL: Here the payload is received by the server as URL part of HTTP request.
- Fragment part of an URL: This part basically contains the portion of the URL separated by '#' symbol from the rest of the URL. Here payload is not received by the server.
- HTML DOM referrer object: The referrer object is the *document.referrer*, which represents currently loaded document's URL. Here the payload is received by the server at the referrer header.

A report by Trustwave's Spiderlabs says that the number of applications that are vulnerable to XSS attack are 82% of the total Web applications (2013)¹. Again, according to WhiteHat Security XSS stood first in the most common vulnerability category (2014)². XSS also tops the list of most frequently occurring vulnerability in the survey carried out by Cenzic (2014)³. CWE by MITRE [8] also warns by saying that XSS is one of the most prevalent, obstinate and dangerous vulnerability in Web application. Among the XSS vulnerabilities, the most frequent one is the reflected XSS attack.

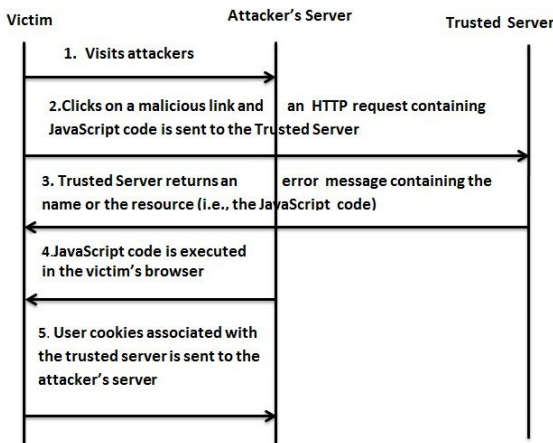
¹<https://www.trustwave.com/Resources/Library/Documents/2013-Trustwave-Global-Security-Report/>

²<http://info.whitehatsec.com/rs/whitehatsecurity/images/statsreport2014-20140410.pdf>

³<http://info.cenzic.com/rs/cenzic/images/Cenzic-Application-Vulnerability-Trends-Report-2013.pdf>



(a) Reflected XSS attack generated through E-mail



(b) Reflected XSS attack generated through the links on malicious user's server

Figure 2: Different scenarios to generate reflected XSS attacks

In this work, we consider the attack instances of reflected XSS attacks. Reflected or Non-persistent XSS attack is the most common attack among all the three XSS attacks. It is easy to create and also can be launched easily to gather sensitive informations. So, attackers generally tend to carry out such attacks frequently. Since the Web service users are the common people who may not have an insight knowledge of the underlying architecture. So it becomes easy for the attackers to trick such individuals by creating a specially crafted URL and making the user to click on that. Moreover, since here the attacker crafted script is reflected back to the user's browser, so one need not to store the script in the server and to wait for user to check that Website for executing the attack.

2.2 Related Work

Javascript has become an unavoidable part of most Web applications due to the necessity of increased interactivity between the user and the Web applications. So the idea of detection of malicious JavaScript is not new. A brief summary of the work done so far, that are thoroughly

studied to understand the present scenario in this field, are mentioned in Table 1.

In [22], Likarish et al. propose a classification based approach for detecting obfuscated malicious JavaScript detection. They propose features that can identify obfuscation, since obfuscation is a well known method of bypassing security filters. Based on the recommendation of various classifiers, the designed malicious JavaScript detector either passes or discards the request.

In [21], Kirda et al. propose a tool *Noxes*. This is a client side solution to mitigate XSS-attack. It uses both manual and auto generated rules to mitigate XSS-attacks. Since we are focusing on designing a solution which balances the loads among server and the client, so it is very important for us to study the existing client based and server based approaches. *Noxes* identifies all the links either as statically embedded or dynamic links. Dynamic links are considered vulnerable to XSS attack, since attacker can embed their code in a dynamic link.

In [33], Wurzinger et al. propose a tool *Secure Web Application Proxy (SWAP)*, which is a server-side solution for mitigating XSS attack. *SWAP* consists of a reverse proxy. It interprets the HTML responses and the modified web browser detects the script contents.

In [23], Di Lucca et al. propose an approach, which is a combination of both static and dynamic approach. Static analysis is used to determine whether a server page is vulnerable to XSS attack. Dynamic approach verifies whether the determined vulnerable web application by static approach is actually vulnerable or not. This approach uses a control flow graph (CFG) to determine the vulnerability in a Web application.

In [28], Salas et al. propose an approach which uses security testing methods like penetration testing and fault injection for detection of XSS attack. Depending upon the results of penetration testing by a user utility referred to as soapUI and interpretation of HTTP status codes in the header of SOAP message, they develop 8 rules. On the basis of which they determine the existence of vulnerability in Web services. Fault injection phase is carried out with *WSInject*, which is placed as proxy between client and server and intercept the messages sent by soapUI before passing it to the server. Faults are injected during this phase. By intercepting the HTTP messages sent by the SOAP request message, they use the previously defined vulnerability analysis rules to determine the injection.

In [2], Athanasopoulos et al. present a tool called *xHunter*, which checks the JavaScript parse tree depth. If the depth is beyond some threshold value, it considers the URL as suspicious.

In [1], Adi et al. propose a design for a proxy named *Wines* that monitors the browser requests sent to a server. Depending upon the patterns of malicious strings kept in different cells of $Wines(T_H1, T_H2)$, the requests are categorized as either harmful or harmless. Harmless strings are forwarded to the server and harmful strings are blocked. All the terms used by the method are biological term since the work is inspired by Human Immune

Table 1: Comparison among existing methods

Referred Work	Year	Description	Dataset(R/S)
Likarish el al. [22]	2009	This technique propose a method to suppress potentially malicious JavaScripts based on the recommendation of classifiers.	S
Kirda el al. [21]	2006	This is a rule based client side solution to mitigate XSS attack.	R
Wurzinger el al. [33]	2009	This server side solution intercepts all HTML responses, and uses a modified Web browser which is utilized to detect script content.	R
Di Lucca el al. [23]	2004	This approach is a combination of static and dynamic approach for detecting XSS attack.	R
Salas et al. [28]	2014	This method is to analyze the robustness of web services by fault injection with WSInject.	R
Athanasopoulos et al. [2]	2010	Proposes a method called xHunter to detect XSS exploits from web trace.	R
Shar et al. [29]	2013	Hybrid model for XSS and SQL injection attack detection	S
Adi et al. [1]	2012	Proposes a method called Wines to detect mutated attack strings.	R
Gupta et al. [11]	2015	Proposed a method to prevent XSS attacks using Apache Tomcat and Web Goat	S
Chun et al. [9]	2016	XSS Attack Detection Method based on Skip List	S

R=Real life, S=Synthetic

System.

Gupta et al. [12] proposed a method called XSS-SAFE for XSS attack detection and prevention based on automated feature injection statements and placement of sanitizers in the injected code of JavaScript. The main advantage of this method is that it can detect XSS attacks without any modification to client- and server-side commodities.

Our approach is a Client-Server based approach, which focuses on balancing the load between client and the server. The detection mechanism in the proxy includes an ensemble based feature selection approach followed by an attribute clustering method to distinguish the malicious traffic from the benign traffic.

3 Proposed Method

The following definitions and theorem provide the theoretical basis of our work. The symbols/notations used to describe our work are reported in Table 2.

Definition 1. Attribute Rank: The rank of an attribute D_{a_i} is defined as the relevance of the attribute a_i for a given class (attack or normal) in a dataset D .

Definition 2. Attribute Cluster: An attribute cluster C_k^i of an attribute D_{a_i} is defined as a subset of objects of a given dataset D (i.e., $C_k^i \subseteq D_{a_i}$) which has high intra-cluster similarity over the attribute D_{a_i} .

Definition 3. Cluster: A cluster C_A is a subset of objects of a given dataset D (i.e., $C_A \subseteq D$) which is obtained by considering the common objects C_A^i over a selected subset of relevant attributes S . In other words,

$$C_A = C_A^1 \cap C_A^2 \cap \dots \cap C_A^S, \quad S \leq n$$

Theorem 1. If $SD_attrib_clus()$ assigns an instance/object O_j to attack group C_A , it cannot be in the normal group of a relevant attribute cluster, w.r.t predefined attribute rank or relevance, i.e., $O_j \notin C_N^i, \forall i = 1, \dots, S$.

Proof: It can be proved by contradiction.

Let an object $O_j \in C_A$ as given by $SD_attrib_clus()$ and also let $O_j \in C_N^i$, i.e., a normal group for a given relevant attribute.

Now, as per definition, the attack group, i.e., C_A given by $SD_attrib_clus()$ is the intersection of all those attribute clusters C_A^i which,

- have high relevance for attack class over selected subset of relevant features and
- have high compactness.

So if $O_j \in C_N^i$, none of the above two conditions are fulfilled. Hence the proof. \square

Table 2: Symbol Table

Symbols Used	Their meaning
D	Dataset.
D_{a_i}	i^{th} Attribute of dataset D ($i = 1, 2, \dots, n$).
C_k^i	Attribute cluster of i^{th} attribute ($k=1,2$).
CP_k^i	Compactness value for cluster C_k^i
S	Subset of attributes.
O_j^i	j^{th} object of i^{th} attribute ($j=1,2,\dots,m$).
n	Total number of attributes
m	Total number of objects
C_A^i	Attack cluster for i^{th} attribute.
C_N^i	Normal cluster for i^{th} attribute.
C_A	Final attack cluster

3.1 Proposed Framework

The proposed framework for the detection of XSS attack is shown in Figure 3. We have proposed a proxy based approach, where it is attempted to balance the load in both the client and the server. A majority of the detection task is carried out in the proxy. An initial check for vulnerability is done in the client side.

A. Client-based Processing: An initial checking for the vulnerability is carried out at the client machine. Though one of our objects is to balance the work load between the client and the server, considering the possible low computational ability of a client, we maintain minimum overhead in the client machine. We assign three tasks to the client, i.e., preprocessing, feature extraction of the captured data and α -divergence test. The client machine also maintains the profiles of attack and normal instances provided by the detection module in the proxy for reference. When the client sends a request to the server, it is handled by the client for preprocessing, feature extraction and α -divergence test with reference to the attack/normal profile. If the value exceeds a pre-defined threshold value then the request is not further processed. It is dropped in the client side only. Otherwise, the request is forwarded to the proxy for further processing.

B. Proxy-level Processing: The majority of the detection tasks are carried out in the proxy server to

keep the load in the main server minimum. This includes a step-by-step method to detect the attack using an unsupervised approach. The method follows four steps in sequence, viz., (a) data gathering, (b) preprocessing and feature extraction, (c) feature selection using an ensemble approach and (d) attack detection using attribute clustering over an optimal subset of relevant features. The steps are discussed in detail next.

B.1 Data Gathering: A major brainstorming task of this proxy level processing is to find the Websites for gathering the attack scripts. Since most of the Websites remove the scripts as they are no longer in use once detected, so finding such scripts are difficult. We have collected most of the attack scripts from [6]. Similarly, we gather the normal scripts using a testbed in our institution.

B.2 Preprocessing and Feature Extraction: This step involves finding out a number of features to describe the gathered data. We have found a total of 15 features relevant to our problem as also can be found in [22]. After extracting the features a 16 dimensional dataset (including the class label) is prepared. But since all the features in the dataset are not equally weighted and the ranges vary by a large margin, we have normalized the dataset using min-max normalization method.

B.3 Feature Selection Using Rank Aggregation: At this step, the features which are least relevant are excluded and only a subset of optimal relevant features is taken. Rank aggregation algorithm available in R package selects an optimal subset of attributes (say S) from total number of attributes n . Rank aggregation framework consists of a number of steps as shown in Figure 4. The prerequisite for the algorithm is a dataset with class labels which is given to different ranking based feature selection algorithms such as infogain [26], correlation based feature selection [14], gain Ratio [25], symmetric uncertainty [13], chi-square [18], mutual information [16] and reliefF [30]. The rankings given by these algorithm are input to a rank aggregation algorithm for the final subset of relevant features generation as shown in the Figure 5.

B.4 Attribute Clustering: Our proposed attribute clustering clusters the instances using the algorithm shown in Algorithm 1. Attribute clustering algorithm is based on the $kmeans$ [15] clustering algorithm. Here each feature is clustered individually applying $kmeans$. The $kmeans$ algorithm refers the parameters,

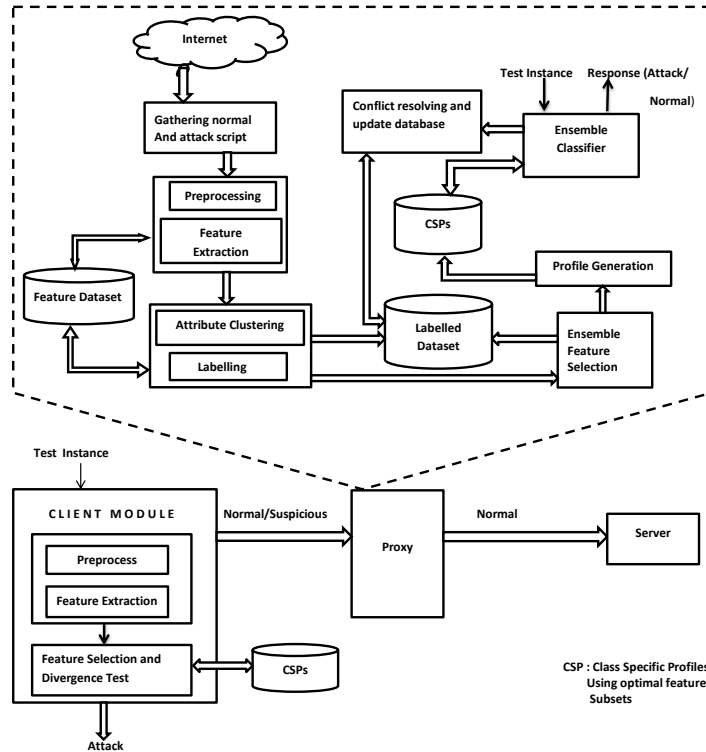


Figure 3: Proposed framework for XSS attack detection

viz., *indexMatrix* and *sumd* (as shown in Algorithm 1). The *indexMatrix* holds the cluster ids for the objects of an attribute. After that, cluster intersection is performed on the basis of the cluster compactness i.e., the more compact cluster of each attribute is considered. The attributes are taken on the basis of their rank given by the rank aggregation algorithm. After clustering, the groups are labeled using supervised approach w.r.t the already built profiles.

3.2 Algorithm for Attribute Clustering

The steps of the proposed attribute clustering algorithm is given in Algorithm 1.

3.3 Complexity Analysis

The complexity of the *SD_attrib_clus()* algorithm is primarily dominated by the *kmeans* clustering algorithm. All other operations are simple merging and intersection operations. So they are of $O(n \times m)$. Where $(n \times m)$ is the dimension of the original dataset. As we know, the complexity of *kmeans* algorithm is $O(n \times m^{(dk+1)} \log m)$. Where $m \times n$ is the dimension of the dataset, d = dimension of the dataset given as input to the *kmeans* algorithm, and k =number of clusters. For our algorithm $k=2$ and $d=1$. Hence the complexity of our algorithm is $O(n \times m^3 \log m)$.

4 Experimental Results

The experiments were carried out in both Windows 7 and Linux environment. The machine used was a 64-bit machine with 2 GB RAM. Matlab 2010 was used to performing attribute clustering. WEKA 3.7.11 was used to run the individual ranking algorithm on the labeled dataset. R package was used to run the rank aggregation algorithm over the rank lists given by the individual rankers. All the experiments carried out can be subdivided into the following sections.

4.1 Dataset Preparation

Dataset preparation involves several steps as described below.

4.1.1 Data Gathering

The first step of dataset generation is the collection of data from the Internet. Since the malicious scripts are immediately removed after detection from the Web applications, so it is very hard to collect live scripts. We have collected attack scripts from [6] and the benign JavaScripts from various Websites, which are using rich JavaScript contents. Figure 6 and Figure 7 are the example of collected attack and normal script respectively.

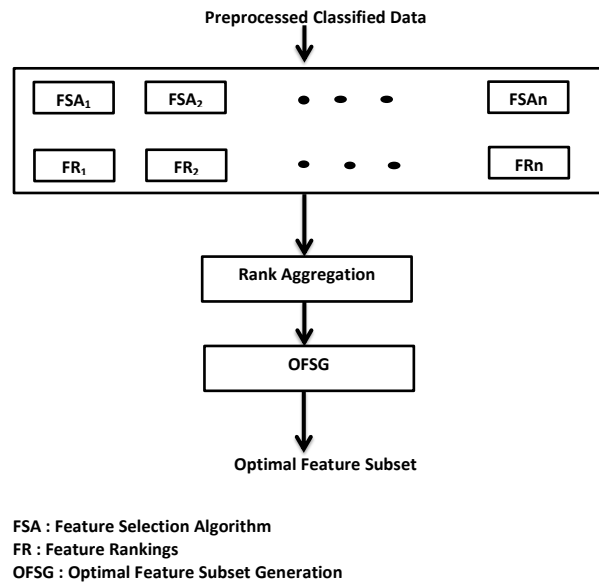


Figure 4: Optimal feature selection framework

4.1.2 Feature Extraction

After gathering the data, the second major phase is to extract the features that are relevant to our problem. After performing a thorough study of existing works [22] and the gathered data we finally picked up fifteen features as described in Table 3.

4.1.3 Modules

After extraction of the fifteen features a total of seventeen dimensional dataset (including the Sl. no. and class label) is prepared. However, while attribute clustering is performed only the first 15 features are considered as shown in Figure 8. The values of the instances for the 16th feature in most cases are found to be zero. The dataset consists of 71 instances as of now and is flexible. That is, at any point of time if we find a new attack script or normal script we can add that instance to the existing dataset.

Different procedures written in C and their functions are described below.

- 1) *extract_script()*: This function extract only the codes included within the script begin tag `<script >` and the script end tag `</script >`. All the codes other then this are discarded as they are not executed as JavaScript. The function also calls all the remaining methods.
- 2) *compute_length()*: Calculates the total number of characters in the script.
- 3) *no_of_lines()*: Calculates the total number of lines in the script.

- 4) *no_of_strings()*: This function outputs the total number of lines in the script.
- 5) *avg_characters()*: It calculates average number of characters per line in the script.
- 6) *percentage_whitespace()*: This method gives the percentage of whitespace characters with respect to the total number of characters in the script.
- 7) *avg_string_length()*: It calculates average length of the strings in terms of number of characters present in the string.
- 8) *no_of_comments()*: This method gives the number of comment lines present in the script.
- 9) *avg_comments_per_line()*: It calculates the average number of comments per line of the script.
- 10) *no_of_words()*: Calculates the total number of words in the script.
- 11) *percentage_of_not_commented_words()*: This method calculates the percentage of words that are not commented over the total number of words present in the script.
- 12) *count_hex_octal()*: It outputs the total number of hexadecimal numbers and octal numbers present in the script.
- 13) *human_readability()*: It gives the output in boolean form i.e., either 'Y'(Yes) or 'N'(No). If a script is human readable then it gives the output as 'Y', otherwise 'N'. Human readability is determined with the help of the following methods.

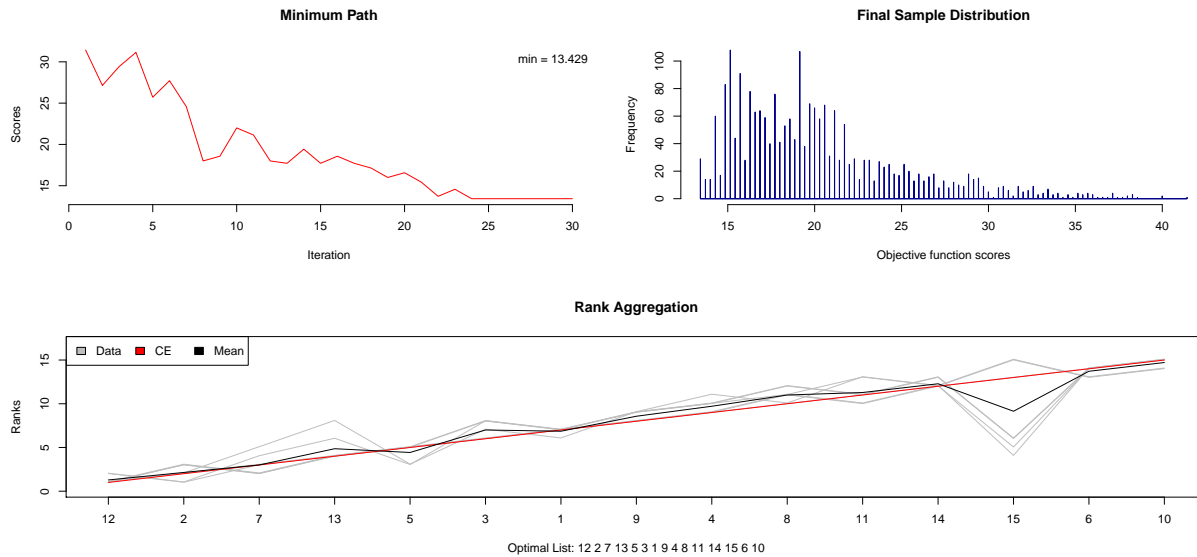


Figure 5: Output of the rank aggregation algorithm with optimal feature subset

- a. *cal_percent_alphabetes()*: Calculates the percentage of words where percentage of alphabets is $>70\%$.
- b. *cal_percent_vowels()*: Calculates percentage of words where percentage of vowels lies in the range $20\%-60\%$.
- c. *percent_length()*: Calculates the percentage of words which are less than 15 characters long.
- d. *percent_repetition()*: Calculates the percentage of words containing repetition of the same letter less than 3 times.

- 14) *methods_called()*: This function gives the total number of methods called in the script.
- 15) *avg_arg_length()*: Calculates the average argument length to each method.
- 16) *count_unicode_char()*: Calculates the number of unicode characters present in the script.

4.1.4 Increase in the Population Density of the Dataset

After collecting the attack and normal scripts, with the help of the modules described in the previous subsection we created a dataset consisting of 71 attack and normal instances in the ratio 1:2 respectively. Now with the help of a module written in C, we increase the number of instances of the dataset to 1078 instances with the same ratio 1:2, respectively. This dataset is used to perform all the operations performed in the following sections.

4.1.5 Normalization of the Dataset

In many pragmatic scenarios, a dataset may consist of attributes or features having values with different ranges.

It generally tends to create problem while the some of the values of some attributes are relatively much larger than that of the other attributes. This is because, larger values have a greater impact on the proximity measures like Euclidean distance. Since the base of our proposed attribute clustering algorithm is kmeans, which uses Euclidean distance measure, so it is very important for us to normalize the dataset. Figure 8 displays a part of the original dataset, whereas Figure 9 shows a part of the dataset after normalization. We have used min-max normalization to normalize the dataset. The formula for which is given next.

$$X_n = \frac{(X - X_{min})}{(X_{max} - X_{min})}$$

Where, X_n = Normalized value between 0 and 1, X = Original value, X_{max} = Maximum value of the attribute, X_{min} = Minimum value of the attribute.

4.2 Results

In this section, we have shown the true positive rate, false positive rate, and accuracy in identifying the groups of attack and normal scripts. An ROC curve is plotted as shown in Figure 10 to demonstrate the detection performance.

4.2.1 ROC Curve

Receiver Operating Curve (ROC) for our dataset is the curve of True Positive Rate (TPR) vs False Positive rate (FPR) of the clusters given by different subset of the attributes or features. Table 4 shows the value of the TPR, FPR, and accuracy of the clusters given by the attribute selection algorithm based on the feature subsets. The feature subsets contains the feature values according to the rank given by the ensemble feature selection algorithm.

```

<script type="text/javascript">

var _gaq = _gaq || [];
_gaq.push(['_setAccount', 'UA-30187030-1']);
_gaq.push(['_trackPageview']);

(function() {
  var ga = document.createElement('script'); ga.type = 'text/javascript';
ga.async = true;
  ga.src = ('https:' == document.location.protocol ? 'https://ssl' :
'http://www') + '.google-analytics.com/ga.js';
  var s = document.getElementsByTagName('script')[0];
s.parentNode.insertBefore(ga, s);
})();

</script>

```

Figure 6: Example of a benign JavaScript

```

<script>
var t="";
var
arr="646f63756d656e742e777269746528273c696672616d65207372
633d22687474703a2f2f766e62
757974612e636f2e62652f666f72756d2e7068703f74703d363735656
1666563343331623166373222
2077696474683d223122206865696768743d223122206672616d656
26f726465723d2230223e3c2f6
96672616d653e2729";for(i=0;i<arr.length;i+=2)t+=String.fromCharCode(
parseInt(arr[i]+arr[i+1],16));eval(t);</script>

```

Figure 7: Example of an attack script

Steps involved in calculating the True Positive (TP) and False Positive (FP) values of a cluster given by the attribute clustering algorithm are as follows:

- The attribute rank subset given by rank aggregation is taken and SD_attrib_clus() algorithm is applied on the whole dataset according to the given feature rank.
- The cluster which is more compact is considered as the attack cluster and the cluster instances are stored in a matrix.
- Now from the actual labeled dataset the attack instances are determined and intersection of these instances with the previously stored cluster instances are found. Thus we get the TP value. And the instances, that are excluded are counted as the FP value.
- The TPR and FPR are calculated from these TP and FP values with the help of the following formulas.

$$\text{True Positive Rate(TPR)} = \frac{\sum \text{True Positive}}{\sum \text{Condition Positive}}$$

$$\text{False Positive Rate(FPR)} = \frac{\sum \text{False Positive}}{\sum \text{Condition Negative}}$$

$$\text{Accuracy(ACC)} = \frac{\sum \text{True Positive} + \sum \text{True Negative}}{\sum \text{Total Instances}}$$

4.3 Comparison with Other Methods

In this section, we compare our method with other competing methods of XSS detection.

- Like [22], our method is also established on feature dataset generated based on the extracted features from attack and normal scripts.
- Like [7, 29], we also evaluate our method in terms of detection accuracy and the performance of our method is highly satisfactory.
- Unlike [7, 22], our method uses unsupervised attribute clustering technique to group the JavaScripts into legitimate and malicious.
- Unlike most other methods [21, 33], our approach attempts to balance the load between the client and server.

Table 3: Description of extracted features

Sl No.	Feature Label	Feature Description
1	A	Number of characters in the script
2	B	Number of lines in the Script
3	C	Number of strings in the script
4	D	Average characters per line
5	E	Percentage of whitespace in the script
6	F	Average string length
7	G	Number of comments in the script
8	H	Average comments per line
9	I	Total number of words
10	J	Percentage of words that are not commented
11	K	Number of octal numbers
12	L	Human readability in terms of yes or no. checking criteria are:
		a)Percentage of words which are >70% alphabetical >=45%
		b)Percentage of words, where 20% < vowels<60% >=40%
		c)Percentage of words which are less than 15 characters long>=70%
		d)Percentage of words containing< 3 repetition of the same letter in a row>=80%
13	M	Number of methods called
14	N	Average argument length
15	O	Number of unicode symbols
16	P	Number of HEX numbers

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
410	5861.6	165.5	1514.8	2115.9	300.8	47.8	29.1	3.3	12595	86	9.9	1	1202.3	8	0.2
411	6337	915.6	1812.6	2480.2	233.1	205.1	10.7	2.8	1940.4	64.1	7.5	1	51.8	12.3	0.6
412	11429.7	1112.4	2933.5	8923.3	44.9	51.2	77.6	0.4	497.4	80.3	7.2	1	160.9	18.9	0.8
413	57262.2	355.7	274.9	4786.2	24.5	53.7	20.9	0.9	10564.2	12.5	5.3	1	320.9	0.7	0.3
414	12317.7	1065.7	950.5	2850.7	34.9	27	39.2	0.5	11461	25	8.4	1	526.7	10.2	0.2
415	44666	576	2548.3	6975.1	255.4	263.7	94.6	2	853.7	74.9	2.6	1	1184.5	12.2	0.2
416	51530.1	906.9	1625.9	8206.6	323.4	240.8	3.6	3	1434.6	69.7	6.2	1	994.7	17.1	0.8
417	15883.1	530.2	1855.9	8459.4	162.3	146.4	41.8	2	3624.7	34.6	3.7	1	228	10.9	0
418	60032.8	1086.6	3114.1	7369.7	307.4	261	54.4	3.7	8477.1	92.2	10.4	1	85.2	13.2	0.9
419	32763.9	103.1	2863.9	8613.7	234.9	81	43.3	3.6	7897.1	17.8	11.6	1	797.5	21.5	0.3
420	33955.9	934.4	241.6	3259.3	287.4	172.1	28.1	1.7	7210.9	68.9	3.3	1	909.5	4.1	0.3
421	49506.8	30.4	683.9	3721.4	119.9	176.7	30.6	3.7	10270.5	51.4	1.5	1	813.6	9.1	0.6
422	26506.8	515.3	3302.4	1709	29	78.1	58.9	2.5	12397.4	78.9	9.8	1	44.7	23.8	0.3
423	3784.2	186.8	2207.5	3449.4	304	263.5	30	2.4	6125.2	11.7	8	1	107.7	5.2	0.1
424	33041.9	237.1	995.5	5424.1	181.8	244.7	22.4	1.8	8937.3	14.7	2.3	1	126.8	10.5	0.2
425	16202.1	98.8	1919.5	639.4	24.5	238.6	65.1	2.2	12439.7	42.3	9.2	1	419.4	19.2	1
426	33070.2	82.2	1980	87.6	366.1	222.5	43.6	2.7	12065.7	61.8	1	1	335.5	6.8	1
427	21530.3	994.2	65.2	3671.8	274.9	187.6	90.6	2.8	1197.2	10	8	1	165.1	14.1	0.9
428	12398.6	190.8	3011	1600.7	369.6	97.8	81.4	3.7	12391.1	70	9.2	1	349.8	15.1	0.3
429	7323.8	733.1	2275.2	7709.7	120.1	169.1	52.8	1.7	8845.1	91.2	0.9	1	718.5	24	0.2
430	45017.3	1013.8	1340.9	6343.9	83.6	68.7	59.9	0.7	12091	63.9	12.3	1	580.3	3.9	0
431	4344.8	986.6	2837	3534.3	175	109.4	77.1	0.6	4022.1	79.6	4.8	1	226.2	14.3	0
432	2407.5	1143.6	2473.3	2373.7	84.2	99.8	42.6	0.6	12773.4	31.5	12.3	1	255.5	22	0.8
433	2351.7	859.1	603.9	4577.9	50.6	263.9	63.1	1.8	9953.2	87.8	1.5	1	474.6	3.7	0.1
434	21437.4	1046.5	1138.2	5105.8	96.9	211.1	69.6	1	887.3	77.4	11.1	1	4.8	14.6	1

Figure 8: A portion of the original dataset

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
410	0.092622	0.142291	0.456045	0.23362	0.806651	0.178476	0.304393	0.825	0.975751	0.860861	0.707143	1	0.9194	0.32	0.2
411	0.100134	0.787206	0.545701	0.273843	0.625101	0.765866	0.111925	0.7	0.150325	0.641642	0.535714	1	0.039612	0.492	0.6
412	0.180606	0.95641	0.883159	0.985238	0.120408	0.191172	0.811715	0.1	0.038534	0.803804	0.514286	1	0.12304	0.756	0.8
413	0.904825	0.30582	0.082761	0.528453	0.065701	0.200508	0.218619	0.225	0.818423	0.125125	0.378571	1	0.245393	0.028	0.3
414	0.194637	0.916258	0.286157	0.314751	0.093591	0.100805	0.410042	0.125	0.887899	0.25025	0.6	1	0.402768	0.408	0.2
415	0.705787	0.495228	0.76719	0.770134	0.684902	0.98469	0.98954	0.5	0.066137	0.74975	0.185714	1	0.905789	0.488	0.2
416	0.814249	0.779726	0.489493	0.906106	0.867257	0.899177	0.037657	0.75	0.11114	0.697698	0.442857	1	0.760648	0.684	0.8
417	0.250976	0.45585	0.558737	0.934018	0.435237	0.546668	0.437238	0.5	0.28081	0.346346	0.264286	1	0.174352	0.436	0
418	0.948604	0.934227	0.93753	0.813702	0.82435	0.974607	0.569038	0.925	0.656732	0.922923	0.742857	1	0.065153	0.528	0.9
419	0.517717	0.088641	0.862205	0.951054	0.629928	0.302451	0.452929	0.9	0.611799	0.178178	0.828571	1	0.609849	0.86	0.3
420	0.536552	0.80337	0.072736	0.359865	0.770716	0.642637	0.293933	0.425	0.558638	0.68969	0.235714	1	0.695496	0.164	0.3
421	0.782278	0.026136	0.205895	0.410887	0.321534	0.659814	0.320084	0.925	0.795669	0.514515	0.107143	1	0.622161	0.364	0.6
422	0.418845	0.443039	0.99422	0.188694	0.077769	0.291622	0.616109	0.625	0.960443	0.78979	0.7	1	0.034182	0.952	0.3
423	0.059796	0.160604	0.664589	0.380854	0.815232	0.983943	0.313808	0.6	0.474527	0.117117	0.571429	1	0.082358	0.208	0.1
424	0.522109	0.203851	0.299705	0.598885	0.48753	0.91374	0.23431	0.45	0.692385	0.147147	0.164286	1	0.096964	0.42	0.2
425	0.256016	0.084944	0.577884	0.070597	0.065701	0.890961	0.680962	0.55	0.96372	0.423423	0.657143	1	0.320716	0.768	1
426	0.522557	0.070672	0.596098	0.009672	0.981765	0.830841	0.456067	0.675	0.934746	0.618619	0.071429	1	0.256557	0.272	1
427	0.34021	0.854784	0.019629	0.40541	0.737195	0.700517	0.947699	0.7	0.092749	0.1001	0.571429	1	0.126252	0.564	0.9
428	0.195916	0.164043	0.906491	0.176736	0.99115	0.365186	0.851464	0.925	0.959955	0.700701	0.657143	1	0.267493	0.604	0.3
429	0.115727	0.630298	0.684971	0.851242	0.32207	0.631434	0.552301	0.425	0.685242	0.912913	0.064286	1	0.549438	0.96	0.2
430	0.711338	0.871636	0.403691	0.700442	0.224189	0.256521	0.626569	0.175	0.936706	0.63964	0.878571	1	0.443756	0.156	0
431	0.068654	0.84825	0.854106	0.390228	0.469295	0.408503	0.806485	0.15	0.311597	0.796797	0.342857	1	0.172975	0.572	0
432	0.038042	0.983234	0.744611	0.262084	0.225798	0.372654	0.445607	0.15	0.989572	0.315315	0.878571	1	0.195381	0.88	0.8
433	0.03716	0.738629	0.18181	0.505454	0.135693	0.985437	0.660042	0.45	0.771088	0.878879	0.107143	1	0.362927	0.148	0.1
434	0.338742	0.899751	0.342666	0.563741	0.259855	0.788271	0.728033	0.25	0.06874	0.774775	0.792857	1	0.003671	0.584	1

Figure 9: A portion of the normalized dataset

Table 4: Accuracy of the classes based on the feature subset

Feature Subset	True Positive Rate(TPR)	False Positive Rate(FPR)	Accuracy
12,2,7,13,5,3,1,9,4,8,11,14,15,6,10	0.24	0	0.7449
12,2,7,13,5,3,1,9,4,8,11,14,15,6	0.517	0.003	0.8358
12,2,7,13,5,3,1,9,4,8,11,14,15	0.978	0.006	0.9889
12,2,7,13,5,3,1,9,4,8,11,14	0.983	0.006	0.9907
12,2,7,13,5,3,1,9,4,8,11	0.983	0.006	0.9907
12,2,7,13,5,3,1,9,4,8	0.983	0.006	0.9907
12,2,7,13,5,3,1,9,4	0.983	0.007	0.9897
12,2,7,13,5,3,1,9	0.983	0.007	0.9897
12,2,7,13,5,3,1	0.983	0.007	0.9897
12,2,7,13,5,3	0.983	0.007	0.9897
12,2,7,13,5	0.983	0.007	0.9897
12,2,7,13	0.983	0.007	0.9897
12,2,7	0.983	0.008	0.9889
12,2	0.983	0.008	0.9889

Data: $D = \text{Dataset}$, $D_{a_i} = i^{\text{th}}$ attribute of D ,
 $\forall i = 1, 2, \dots, n$

$K = \text{No. of clusters}$

Result: $C_A = \text{Attack cluster}$

Function $\text{SD_attrib_clus}()$

```

foreach attribute  $D_{a_i} \in D$  do
  [indexMatrix, sumd] = kmeans( $D_{a_i}$ ,  $K$ )
  foreach Cluster  $C_k^i, k = 1, 2$  do
    |  $CP_k^i = \frac{\text{sumd}}{\text{No. of objects in } C_k^i}$ 
  end
  if  $CP_k^i < CP_{k+1}^i, k=1$  then
    | foreach Object  $O_j^i \in D_{a_i}$  do
      | if  $\text{indexMatrix}(O_j^i) == k$  then
        | |  $C_A^i \leftarrow O_j^i$ 
        | end
      | else
        | |  $C_N^i \leftarrow O_j^i$ 
        | end
      | end
    | end
  end
  else
    | foreach Object  $O_j^i \in D_{a_i}$  do
      | if  $\text{indexMatrix}(O_j^i) == k+1$  then
        | |  $C_A^i \leftarrow O_j^i$ 
        | end
      | else
        | |  $C_N^i \leftarrow O_j^i$ 
        | end
      | end
    | end
  end
end
// Find the common objects of the attributes
  in the order given by rank aggregation
  method
 $C_A = \cap C_A^i, \forall i = 1, 2, \dots, S, S \leq n$ 
Algorithm 1: Attribute clustering algorithm

```

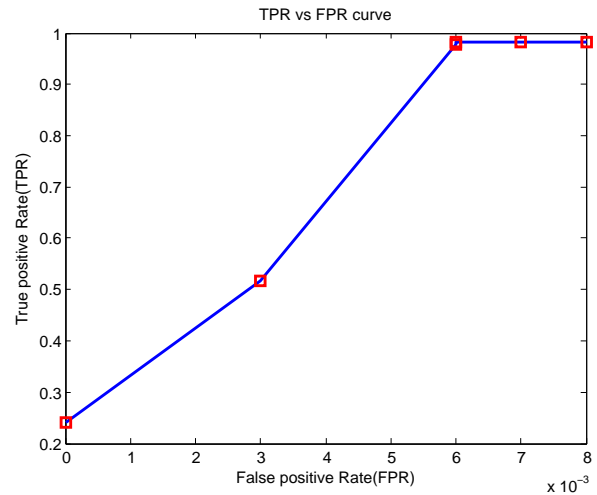


Figure 10: ROC curve

5 Discussion and Conclusion

Based on the results we got from the attribute clustering algorithm, proceeded by rank aggregation using cross entropy monte carlo algorithm, shows us a way how we can use unsupervised techniques in clustering the malicious and benign scripts into two classes with high accuracy. The computation overhead also decreases significantly in the proxy as our proposed method distributes the task between the client and the server. The detection mechanism in the proxy is easy to implement and requires a little knowledge to detect an attack with high accuracy.

Acknowledgments

The authors would like to thank the Ministry of HRD, Govt. of India for funding as a Centre of Excellence with thrust area in Machine Learning Research and Big Data Analytics for the period 2014-2019.

References

- [1] E. Adi, "A design of a proxy inspired from human immune system to detect SQL injection and cross-site scripting," *Procedia Engineering*, vol. 50, pp. 19–28, 2012.
- [2] E. Athanasopoulos, A. Krithinakis, and E. P. Markatos, "Hunting cross-site scripting attacks in the network," in *Third International Conference on Advanced Computing (ICoAC'11)*, pp. 89–92, 2011.
- [3] D. Balzarotti, M. Cova, V. Felmetzger, N. Jovanovic, E. Kirda, C. Kruegel, and G. Vigna, "Saner: Composing static and dynamic analysis to validate sanitization in web applications," in *IEEE Symposium on Security and Privacy (SP'08)*, pp. 387–401, 2008.
- [4] D. K. Bhattacharyya and J. K. Kalita, *Network anomaly detection: A machine learning perspective*, CRC Press, 2013.

- [5] M. H. Bhuyan, D. K. Bhattacharyya, and J. K. Kalita, "Survey on incremental approaches for network anomaly detection," *International Journal of Communication Networks and Information Security*, vol. 3, no. 3, pp. 226–239, 2011.
- [6] Blwood, *Multiple XSS Vulnerabilities in Tikiwiki 1.9.x*, 2006. (<http://www.securityfocus.com/archive//435127>)
- [7] D. Canali, M. Cova, G. Vigna, and C. Kruegel, "Prophiler of a fast filter for the large-scale detection of malicious web pages," in *Proceedings of the 20th International Conference on World Wide Web*, pp. 197–206, 2011.
- [8] S. Christey, *2011 CWE/SANS Top 25 Most Dangerous Software Errors*, 2011. (<http://cwe.mitre.org/top25>)
- [9] S. Chun, C. Jing, H. ChangZhen, X. JingFeng, W. Hao, and M. Raphael, "A xss attack detection method based on skip list," *International Journal of Security and Its Applications*, vol. 10, no. 5, pp. 95–106, 2008.
- [10] J. Grossman, *XSS Attacks: Cross-site scripting exploits and defense*, Syngress, 2007.
- [11] B. B. Gupta, S. Gupta, S. Gangwar, M. Kumar, and P. K. Meena, "Cross-site scripting (XSS) abuse and defense: exploitation on several testing bed environments and its defense," *Journal of Information Privacy and Security*, vol. 11, no. 2, pp. 118–136, 2015.
- [12] S. Gupta and B. B. Gupta, "XSS-SAFE: a server-side approach to detect and mitigate cross-site scripting (XSS) attacks in javascript code," *Arabian Journal for Science and Engineering*, vol. 41, no. 3, pp. 897–920, 2016.
- [13] M. A. Hall, *Correlation-based Feature Selection for Machine Learning*, Doctoral Dissertation, The University of Waikato, 1999.
- [14] M. A. Hall and L. A. Smith, "Feature selection for machine learning: Comparing a correlation-based filter approach to the wrapper," in *Proceedings of the Twelfth International Florida Artificial Intelligence Research Society Conference*, pp. 235–239, 1999.
- [15] J. A. Hartigan and M. A. Wong, "Algorithm as 136: A k-means clustering algorithm," *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, vol. 28, no. 1, pp. 100–108, 1979.
- [16] N. Hoque, D. K. Bhattacharyya, and J. K. Kalita, "MIFS-ND: a mutual information-based feature selection method," *Expert Systems with Applications*, vol. 41, no. 14, pp. 6371–6385, 2014.
- [17] N. Hoque, D. K. Bhattacharyya, and J. K. Kalita, "Botnet in ddos attacks: trends and challenges," *IEEE Communications Surveys & Tutorials*, vol. 17, no. 4, pp. 2242–2270, 2015.
- [18] L. Huan and R. Setiono, "Chi2: feature selection and discretization of numeric attributes," in *Proceedings of Seventh International Conference on Tools with Artificial Intelligence*, pp. 388–391, 1995.
- [19] T. Jim, N. Swamy, and M. Hicks, "Defeating script injection attacks with browser-enforced embedded policies," in *Proceedings of the 16th International Conference on World Wide Web*, pp. 601–610, New York, NY, USA, 2007.
- [20] N. Jovanovic, C. Kruegel, and E. Kirda, "Pixy: a static analysis tool for detecting web application vulnerabilities," in *IEEE Symposium on Security and Privacy*, pp. 6–263, 2006.
- [21] E. Kirda, C. Kruegel, G. Vigna, and N. Jovanovic, "Noxes: A client-side solution for mitigating cross-site scripting attacks," in *Proceedings of the 2006 ACM Symposium on Applied Computing*, pp. 330–337, New York, NY, USA, 2006.
- [22] P. Likarish, J. Eunjin, and J. Insoon, "Obfuscated malicious javascript detection using classification techniques," in *4th International Conference on Malicious and Unwanted Software (MALWARE'09)*, pp. 47–54, 2009.
- [23] G. A. Di Lucca, A. R. Fasolino, M. Mastroianni, and P. Tramontana, "Identifying cross site scripting vulnerabilities in web applications," in *26th Annual International Telecommunications Energy Conference*, pp. 71–80, 2004.
- [24] Y. Minamide, "Static approximation of dynamically generated web pages," in *Proceedings of the 14th International Conference on World Wide Web*, pp. 432–441, New York, NY, USA, 2005.
- [25] J. R. Quinlan, *C4. 5: Programs for Machine Learning*, Elsevier, 2014.
- [26] D. Roobaert, G. Karakoulas, and N. V. Chawla, "Information gain, correlation and support vector machines," in *Feature Extraction*, pp. 463–470, 2006.
- [27] S. Saha, "Consideration points detecting cross-site scripting," *International Journal of Computer Science and Information Security*, vol. 4, no. 1 & 2, Aug. 2009.
- [28] M. I. P. Salas and E. Martins, "Security testing methodology for vulnerabilities detection of XSS in web services and ws-security," *Electron Notes in Theoretical Computer Science*, vol. 302, pp. 133–154, 2014.
- [29] L. K. Shar, H. B. K. Tan, and L. C. Briand, "Mining sql injection and cross site scripting vulnerabilities using hybrid program analysis," in *Proceedings of International Conference on Software Engineering*, pp. 642–651, Piscataway, NJ, USA, 2013.
- [30] Y. Wang and F. Makedon, "Application of relief-f feature filtering algorithm to selecting informative genes for cancer classification using microarray data," in *IEEE Computational Systems Bioinformatics Conference*, pp. 497–498, 2004.
- [31] G. Wassermann and Z. Su, "Static detection of cross-site scripting vulnerabilities," in *Proceeding of ACM/IEEE 30th International Conference on Software Engineering*, pp. 171–180, 2008.
- [32] D. Wichers, *OWASP, The Open Web Application Security Project*, 2013. (<http://www.owasp.org>)

- [33] P. Wurzinger, C. Platzer, C. Ludl, E. Kirda, and C. Kruegel, "SWAP: Mitigating XSS attacks using a reverse proxy," in *Proceeding of 5th International Workshop on Software Engineering for Secure Systems*, IEEE Computer Society, 2009.

Biography

Swaswati Goswami obtained Master of Technology degree in Information Technology from Tezpur University, India in the year 2012. Her research interests are machine learning and network security.

Nazrul Hoque obtained Master of Technology degree in Information Technology from Tezpur University, India in the year 2012. Currently, he is a PhD candidate in the Department of Computer Science and Engineering at Tezpur University. His research interests are machine learning and network security.

Dhruba K Bhattacharyya received his Ph.D. in Computer Science from Tezpur University in 1999. He is a Professor in the Computer Science & Engineering Department at Tezpur University. His research areas include data mining, bioinformatics, network security, and big data analytics. Prof. Bhattacharyya has published 220+ research papers in the leading international journals and conference proceedings. In addition, Dr Bhattacharyya has written/edited 8 books. His book on Network Anomaly Detection: A Machine Learning Perspective is now popular among the network security researchers. Professor Bhattacharyya is Project Investigator of several prestigious major research grants such as Ministry of HRD's Center of Excellence under FAST, Center of High Performance Computing and UGC SAP DRS II of Govt. of India.

Medical Image Encryption Scheme Based on Arnold Transformation and ID-AK Protocol

Osman Wahballa^{1,2}, Abubaker Wahaballa¹, Fagen Li¹, Idris Ibn Idris³ and Chunxiang Xu¹
(Corresponding author: Abubaker Wahaballa)

University of Electronic Science and Technology of China, Chengdu, China¹

(Email: wahaballah@hotmail.com)

Karary University, Khartoum, Sudan²

Modibbo Adama University of Technology, Yola, Nigeria³

(Received May 10, 2016; revised and accepted Sept. 3 & Oct. 6, 2016)

Abstract

Providing security on transmitted medical image over public channels has become an essential part of computer-aided diagnosis systems. In this paper, we propose an efficient image encryption scheme for medical applications based on Arnold transformation and pairing-free identity-based authenticated key agreement protocol. This allows user to send and receive medical images over public channel safely, while maintaining patient privacy. We then provide the numerical analysis results to prove the robustness of our scheme. These results are carried out via both theoretic analysis and experimental simulations based on MATLAB. The analysis demonstrates that our scheme meets the effectiveness and security requirements of image encryption.

Keywords: Arnold Transformation; Identity (ID)-based Cryptography; Medical Image Encryption Scheme; Statistical Attack

1 Introduction

Image-based diagnostics has become an effective tool for the treatment and prediction of many diseases. In health-care system, the medical images can be transmitted across public channels such as the Internet. However, these images contain very sensitive and confidential information. Therefore, maintaining security and confidentiality of medical images is of utmost priority. Most of the current medical images protection techniques use symmetric encryption [2, 4, 34], traditional public key cryptography [17, 32] or watermarking [13, 19]. However, symmetric encryption suffers from the problem that the same key must be shared by the sender and the receiver and traditional PKC has a complex certificate management, while watermarking is lacking of a standard attack benchmark and distortion measurement [28]. Image encryption techniques are classified based on both spatial and frequency

domain [30]. Arnold transformation has been adopted in a wide variety of multimedia securities because of its periodicity.

Identity(ID)-based cryptography aims to simplify the complex certificate management in the traditional PKC by deriving user's public key from his/her identity. The major advantage of IBC is that it does not require the use of digital certificates to guarantee the authenticity [29]. *Key-agreement protocol* is process whereby two or more parties can establish a shared secret key in such a way that both sides agree with the outcome. Identity-based authenticated key agreement is a useful cryptographic primitive and has been widely used in various applications. In cryptography, bilinear pairing is a mathematical function which combine elements of two cryptographic groups to a third group. Bilinear pairing is widely used to construct or analyze various kinds of authenticated key agreement protocols. However, a bilinear pairing operation is more time-consuming than other operations over elliptic curve group.

In this paper, we propose an encryption scheme for the medical image by incorporating the idea of identity-based authenticated key agreement and Arnold transformation.

1.1 Motivations

Providing security on transmitted medical image has become more and more important with rapid development of both image-based diagnostics techniques and Internet in the field of medical informatics. Furthermore, Health Insurance Portability and Accountability Act (HIPAA) [23] issued mandates for ensuring privacy and security of electronic health information, where healthcare providers are obliged to take appropriate safeguards and measures to ensure that patient information is only provided to people who have a professional need. To reap the benefits of ehealth by achieving better health outcomes, and to improve healthcare quality and efficiency, healthcare providers and patients alike must trust that the patient's

health information is private and secure. The goal of this work is to create an encryption scheme for the medical image by combining identity-based encryption with and an Arnold transformation.

In a nutshell, our contribution is threefold:

- An efficient encryption scheme for the medical image from identity-based encryption is presented that allows user to send and receive medical images over public channels safely, while maintaining patient's privacy and confidentiality.
- A pairing-free identity-based key exchange protocol for medical image encryption is introduced.
- Numerical analysis results are carried out to prove the robustness of our scheme. The analyses demonstrate that our scheme meets the effectiveness and security requirements of image encryption.

The remainder of this paper is organized as follows. In the next section, the state-of-the-art is discussed. Section 3 presents the preliminaries of this paper. The proposed scheme is introduced and discussed in Section 4, while Section 5 is devoted to experimental results. Finally, we conclude the paper in Section 6.

2 State-of-the-Art

Nowadays designing a secure and efficient encryption schemes is a crucial issue for digital image encryption. Due to the large image size, conventional cryptosystems are widely used, such as RSA [22], however, it cannot easily be directly used for image encryption. Instead of the above solution, some researchers focus on designing symmetric image cryptosystems. In particular, a number of schemes [5, 8, 9, 12, 21, 25, 26, 33], based on chaos have been proposed. The chaos-based cryptosystems has some inherent features, such as sensitivity to initial condition and pseudo randomness, therefore, this solution appear more suitable for high-security encryption. Nevertheless, chaos-based schemes have their own weaknesses in terms of exchanging and distributing the symmetric secret keys. This is a particularly serious problem due to the large number of users. In addition, the solution based on chaos-based cryptosystems may have unknown vulnerabilities. Recently several image encryption algorithms founded on chaos have been broken [1, 15, 16, 24]. For instance, an encryption scheme based on improved hyper chaotic sequences is addressed by C. Zhu [33]. Their scheme used a four-dimensional hyper-chaos system in order to generate a pseudo-random number sequence. Later the sequence is applied to control the modulation addition and the bit-wise exclusive OR operation. C. Li et al. [16] analyzed that; if two known plain-images and the corresponding cipher-images are available this scheme can be easily broken. Ideally, in order to avoid these problems a public key encryption is highly recommended. The public key encryption for large image based on elliptic curve is considered by L. Chen et al [6].

3 Preliminaries

In this section, we describe the basic definitions and assumptions that are used in our scheme.

3.1 Elliptic Curves Cryptography (ECC)

The ECC was proposed by Miller and Koblitz [14, 18] as an alternative to RSA in public key cryptography. Any cryptosystem based on ECC provides high security with small key size, for example, a 160-bit ECC is considered to be as secured as 1024-bit RSA key [11]. Let F_q be a field of integers of a modulo a large prime number q . A non-singular elliptic curve $E_q(a, b)$ over F_q is defined by the following equation

$$y^2 \bmod q = (x^3 + ax + b) \bmod q, \quad (1)$$

where $a, b, x, y \in F_q$ with the discriminant $\Delta = (4a^2 + 27b^2) \bmod q \neq 0$. A point $P(x, y)$ is an elliptic curve point if it satisfies Equation (1), and the point $Q(x, -y)$ is called the negative of P , i.e. $Q = -P$. The points $E_q(a, b)$ together with a point \mathcal{O} (called point at infinity) form an additive cyclic group G_q , that is, $G_q = \{(x, y) : a, b, x, y \in F_q \text{ and } (x, y) \in E_q(a, b)\} \cup \{\mathcal{O}\}$ of prime order q . Scalar multiplication over $E|F_q$ can be computed as follows:

$$tP = P + P + \dots + P \quad (t \text{ times}). \quad (2)$$

3.2 Arnold Transformation

The Arnold transformation, also referred to as cat map, is one of the images scrambling techniques that was named after the Russian mathematician Vladimir Arnold, who demonstrated its effectiveness in image processing. The Arnold transformation is periodic, besides it can only be used with square images. The general form of Arnold transformation appears in Equation (3). This equation can be adopted for digital images as follows. Let (i, j) be pixel for $N \times N$ digital image $Img[i][j]$. This image is transformed to $Img[i'][j']$ using Equation (3). As mentioned, Arnold transformation is periodic with period T that depends on the size of the images. Due to the periodicity equation 3 is applied 3 in both scrambling as descrambling processes. If the Arnold is applied (t) times to yield a scrambled image in the sender side, it should be applied $(T - t)$ times to yield the original image in the receiving side. Arnold transformation period T is given by Equation (4) [10, 31].

$$\begin{bmatrix} i' \\ j' \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 1 & 2 \end{bmatrix} \begin{bmatrix} i \\ j \end{bmatrix} \bmod N \quad (3)$$

$$T = 1.4938N + 40.8689, \quad \text{where } 2 \leq N \leq 2000 \quad (4)$$

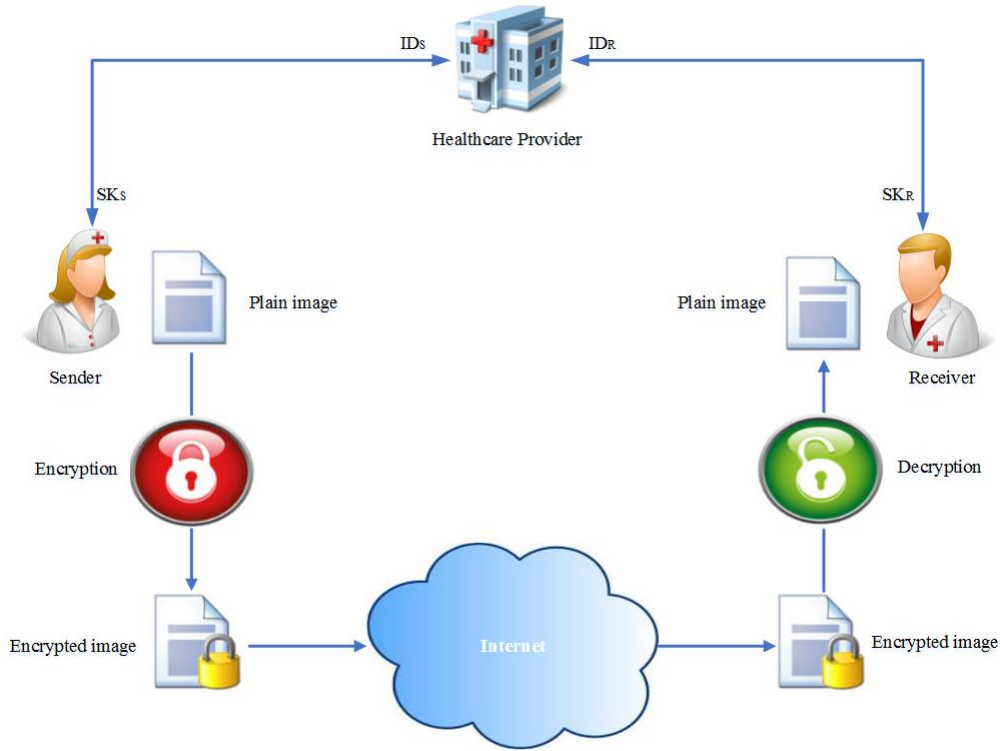


Figure 1: System structure

Table 1: Notations of our model

\mathcal{HP}	Healthcare Provider
\mathcal{S}	Sender
\mathcal{R}	Receiver
PP, S_o	Public Parameters and Master Key
ID_S	Sender's Identity
ID_R	Receiver's Identity
SK_S	Sender's secret key
SK_R	Receiver's secret key
sk	Shared secret key
Img_p	A plain medical image
Img_ϕ	An encrypted medical image
H_1, H_2	Two hash functions

The interaction scenario between the above entities is divided into four phases: *Setup and Registration Phase*, *Key Agreement Phase*, *Encryption Phase* and *Decryption Phase*. Figure 1 shows the sketch of the interaction scenario.

In the setup and registration phase, \mathcal{HP} inputs the security parameters as defined in Section 3.1. Then, it generates the public parameters $params$ and a master key s . Further, sender \mathcal{S} and receiver \mathcal{R} with identities ID_S and ID_R register at \mathcal{HP} . Afterward, \mathcal{HP} generates sender's and receiver's secret keys, SK_S and SK_R respectively. In key agreement phase, \mathcal{S} and \mathcal{R} establish an authenticated session key. Using the shared secret key K and Arnold transformation, the sender \mathcal{S} encrypts a plain medical image IMG_p to get an encrypted image IMG_c , and finally sends it over Internet to the receiver \mathcal{R} . Upon receiving the encrypted image IMG_c , \mathcal{R} uses the shared secret sk to decrypt it.

For convenience, the notations of the proposed scheme are defined in Table 1.

4 Proposed Scheme

4.1 Overview of Our Scheme

In this section, we describe our scheme in the high level. The proposed scheme consists of three entities: a sender \mathcal{S} , receiver \mathcal{R} and healthcare provider \mathcal{HP} . \mathcal{HP} is adopted as trusted third party in our scheme. It is responsible to initialize the public system parameters. In order to establish a secure communication channel between \mathcal{S} and \mathcal{R} , we employ identity-based key exchange protocol [3].

4.2 Concrete Construction

In this section, we concretely construct a medical image encryption scheme by incorporating Arnold transformation and identity-based key exchange protocol. The proposed scheme is composed of the following phases.

4.3 Setup and Registration Phase

4.3.1 Setup

Initially, \mathcal{HP} inputs the security parameters k and determines the tuple $\{F_q, E|F_q, G, P\}$ as defined in Section 3.1. Then, it picks secret master key $\alpha \in \mathbb{Z}_q^*$ and computes its public master-key $S_o = \alpha P$. Afterward, \mathcal{HP} chooses two hash functions $H_1 : \{0, 1\}^* \times G \rightarrow \mathbb{Z}_q$ and $H_2 : \{0, 1\}^* \times \{0, 1\}^* \times G \times G \times G \times G \rightarrow \{0, 1\}^k$. Finally, the \mathcal{HP} publishes the system parameters: $PP = (F_q, E|F_q, G, P, S_o, H_1, H_2)$.

4.3.2 Registration

The sender \mathcal{S} with identity ID_S and receiver \mathcal{R} with identity ID_R register at \mathcal{HP} . Given user's identity ID_i , public parameters PP and public master key S_o , \mathcal{HP} picks $r \in \mathbb{Z}_q^*$, and computes $R_i = rP$ and $H_1(ID_i || R_i)$. Then, it computes $S_i = r + h_i \alpha$. The \mathcal{HP} sets the pair (S_i, R_i) as user's long-term private key. The pair (S_i, R_i) is transmitted to the user U_i secretly. U_i check if $S_i P = R_i + H_1(ID_i || R_i) S_o$ holds. If it does, the long-term private key is valid, reject otherwise.

4.4 Key Agreement Phase

In this phase, sender \mathcal{S} and receiver \mathcal{R} establish an authenticated session key as follows.

Step 1: \mathcal{S} chooses at random the ephemeral key $s \in_R \mathbb{Z}_q^*$ and computes the key token $T_S = sP$.

Step 2: \mathcal{S} sends $M_S = (R_S, T_S, ID_S)$ to \mathcal{R} .

Step 3: Upon \mathcal{R} receiving M_S , he chooses \mathcal{R} 's ephemeral key $r \in_R \mathbb{Z}_q^*$ and computes the key token $T_R = rP$.

Step 4: \mathcal{R} sends $M_R = (R_R, T_R, ID_R)$ to \mathcal{S} .

Step 5: Then, both sides can compute the shared secrets as follows:

- \mathcal{S} computes

$$K_{SR}^1 = S_S T_R + s(R_R + H_1(ID_R || R_R)) S_o$$

and $K_{SR}^2 = s T_R$.

- \mathcal{R} computes

$$K_{RS}^1 = S_R T_S + r(R_S + H_1(ID_S || R_S)) S_o$$

and $K_{RS}^2 = r T_S$.

Step 6: Eventually, \mathcal{S} and \mathcal{R} can compute the shared secret keys as:

$$\begin{aligned} sk &= H_2(ID_S || ID_R || T_S || T_R || K_{SR}^1 || K_{SR}^2) \\ &= H_2(ID_S || ID_R || T_S || T_R || K_{RS}^1 || K_{RS}^2). \end{aligned}$$

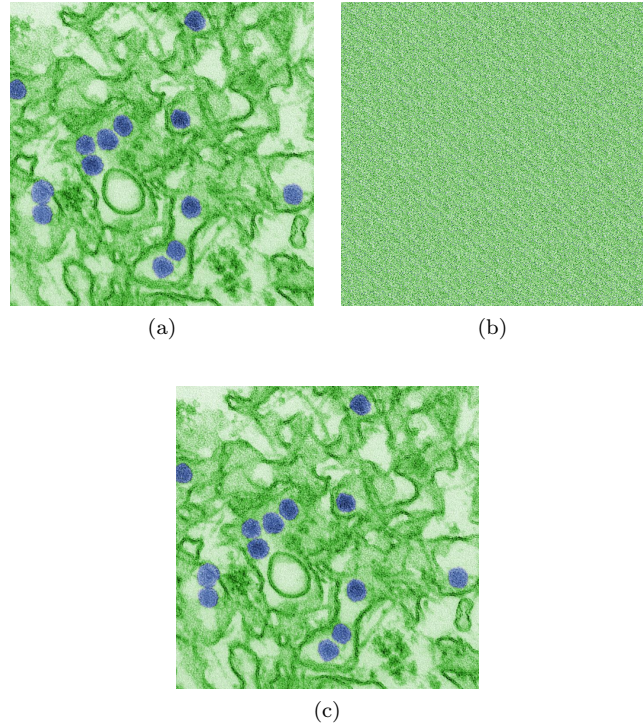


Figure 2: (a) Zika virus original image; (b) Zika virus encrypted image; (c) Zika virus decrypted image.

4.5 Encryption Phase

In this phase, sender \mathcal{S} encrypts a plain medical image Img_ρ using the shared secret key and Arnold transformation algorithm to get the encrypted image Img_ϕ as: $Img_\phi = \Phi_{sk}(Img_\rho)$, where Φ is the Arnold transformation scrambling algorithm. The pseudo-code of scrambling process is illustrated in Algorithm 1. From steps 1-5, algorithm parameters are initialized. We perform scrambling process in steps 6-13 within Arnold transformation period T . In steps 14-21, we calculate best scrambling iteration Bst_τ , which has a minimum correlation coefficient. The Bst_τ is used as descrambling period in the next phase.

4.6 Decryption Phase

Upon receiving the encrypted image Img_ϕ from the sender \mathcal{S} , receiver \mathcal{R} uses the shared secret key sk and Arnold transformation descrambling algorithm to decrypt the Img_ϕ as: $Img_\rho = \Psi_{sk}(Img_\phi)$. where Ψ is the Arnold transformation descrambling algorithm. Due to the periodicity of Arnold transformation, same steps 1-13 in algorithm 1 are applied for descrambling process, where T is replaced by Bst_τ .

5 Experimental Results

The scope of this section is to present an experimental result of our proposed scheme. We use the following medical images with two different image sizes 512×512 and

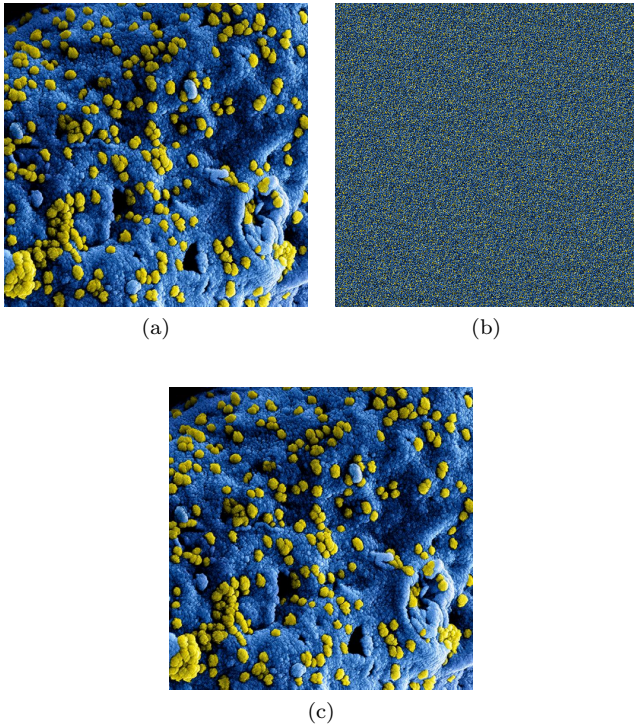


Figure 3: (a) MERS-CoV original image; (b) MERS-CoV encrypted image; (c) MERS-CoV decrypted image.

Algorithm 1: Arnold transformation scrambling algorithm Φ

Input: Img_ρ // plain image
Output: Img_ϕ // encrypted image

```

1  $Img_\rho \leftarrow Img_\phi$ 
2  $T \leftarrow 1.4938N + 40.8689$  // Arnold transform
   period as in Equation (4)
3  $t \leftarrow 0$ 
4  $w \leftarrow Img_\rho.width$ 
5  $h \leftarrow Img_\rho.height$ 
6 while  $t < T$  do
7   for  $i \leftarrow 0$  to  $w$  do
8     for  $j \leftarrow 0$  to  $h$  do
9        $pixel \leftarrow Img_\rho[i][j]$ 
10       $Img_\phi[(2 * i + i) \bmod w][(i + j) \bmod h] \leftarrow$ 
          $pixel$ 
11  $c[t] \leftarrow \sigma(Img_\rho, Img_\phi)$  // calculate the
     correlation coefficient
12  $t \leftarrow t + 1$ 
13 return  $Img_\phi$ 
14 for  $m \leftarrow 0$  to  $T - 1$  do
15    $count \leftarrow 0$ 
16   for  $n \leftarrow 0$  to  $T - 1$  do
17     if  $(c[m] < c[n])$  then
18        $count \leftarrow count + 1$ 
19   if  $(count = T - 1)$  then
20     break
21    $Bst_\tau \leftarrow m$  // Best iteration
    
```

Table 2: Experimental results of entropy analysis.

Image	Image status	Image size (KB)	entropy
Zika virus	Original 512×512	151.552	7.6145
	Encrypted 512×512	192.512	7.6072
	Decrypted 512×512	154.723	7.5904
	Original 256×256	40.960	7.5543
	Encrypted 256×256	49.783	7.5412
	Decrypted 256×256	42.152	7.5378
MERS-CoV	Original 512×512	163.840	7.7038
	Encrypted 512×512	200.704	7.6889
	Decrypted 512×512	178.254	7.6813
	Original 256×256	45.056	7.7016
	Encrypted 256×256	53.248	7.6946
	Decrypted 256×256	46.122	7.6865
TBRF	Original 512×512	73.728	7.9322
	Encrypted 512×512	143.360	7.8571
	Decrypted 512×512	96.364	7.7591
	Original 256×256	24.576	7.8814
	Encrypted 256×256	36.864	7.7802
	Decrypted 256×256	32.523	7.7791

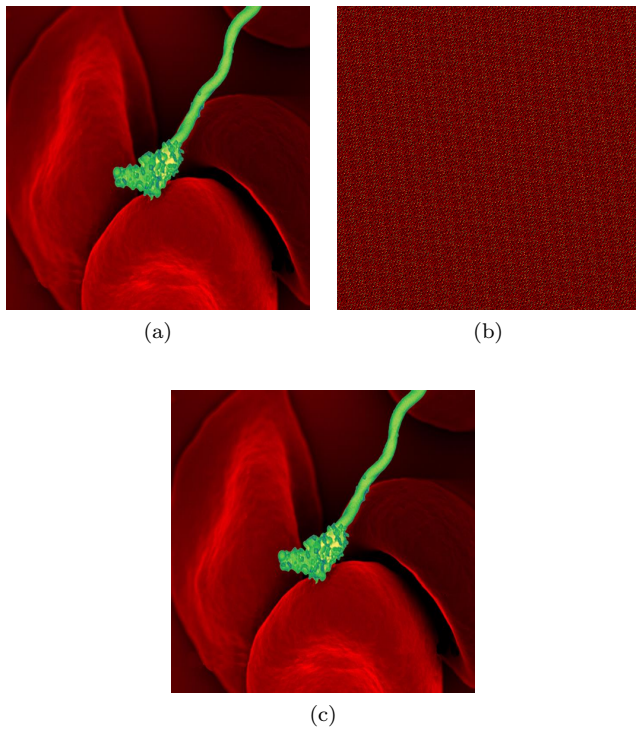


Figure 4: (a) TBRF original image; (b) TBRF encrypted image; (c) TBRF decrypted image.

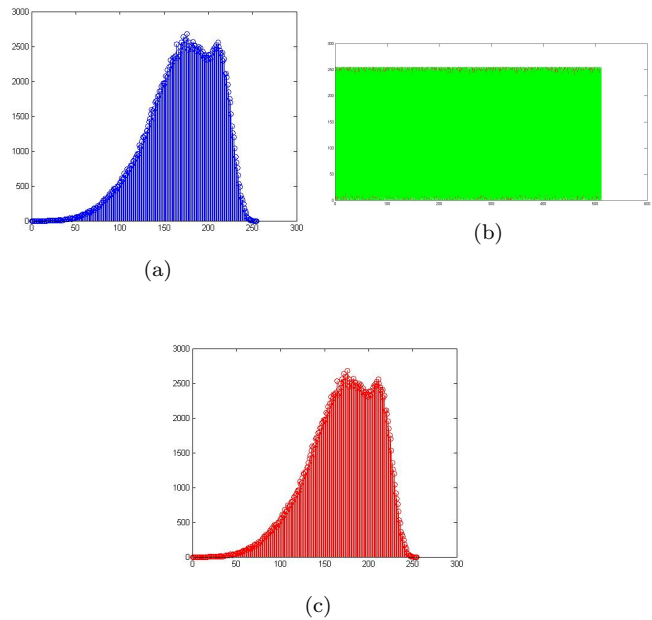


Figure 5: (a) Histogram of original Zika virus image ; (b) Histogram of encrypted Zika virus image; (c) Histogram of decrypted Zika virus image.

256 × 256 24-bits:

- 1) Colorized image shows particles of Zika virus, which is a member of the family Flaviviridae. The virus particles are colored blue in the picture;
- 2) Colorized SEM showing numerous Middle East respiratory syndrome Coronavirus (MERS-CoV) viral particles (yellow) on the surface of a Vero E6 cell (blue);
- 3) Colorized SEM of a spiral-shaped Borrelia hermsii bacterium (green) on a number of red-colored red blood cells. B. hermsii is the causative agent of tick-borne relapsing fever (TBRF).

Figures 2, 3 and 4 show the above images and their corresponding encrypted and decrypted images respectively. Image encryption techniques aim to reduce the correlation of pixel positions and values until they are irrelevant to each other. Therefore, measurement tools used in this evaluation include entropy analysis and correlation coefficients. The entropy is given by Equation (5).

$$H(P) = - \sum_{i=1}^n \sum_{j=1}^n P(x_i, y_j) \log_2 P(x_i, y_j) \quad (5)$$

where, $P(x_i, y_j)$ is the probability of pixel with coordinates (x_i, y_j) in original image appearing at the $[i^{th}][j^{th}]$ blocks in the scrambled image. As indicated in Table 2, the values of entropy analyses for all images are very close

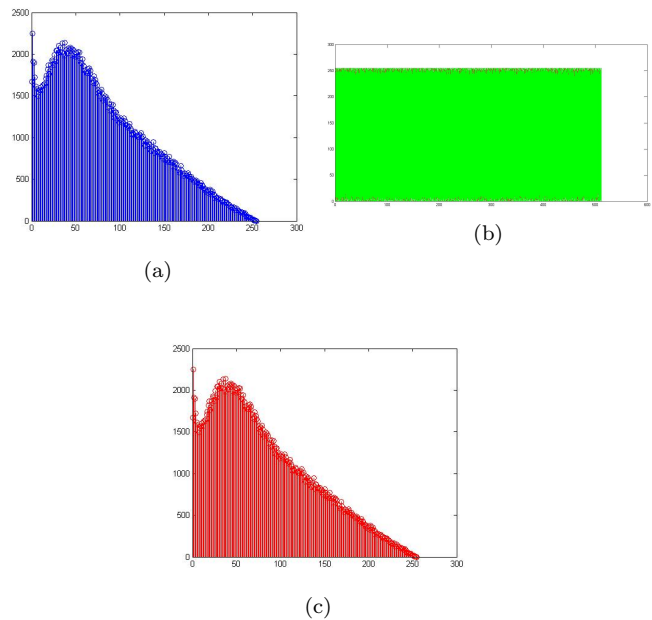


Figure 6: (a) Histogram of original MERS-CoV image ; (b) Histogram of encrypted MERS-CoV image; (c) Histogram of decrypted MERS-CoV image.

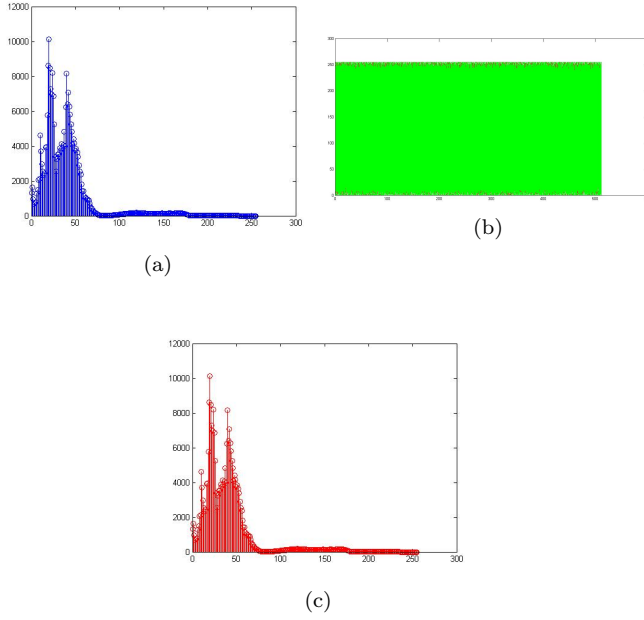


Figure 7: (a) Histogram of original TBRF image ; (b) Histogram of encrypted TBRF image; (c) Histogram of decrypted TBRF image.

to the ideal value 8 [7]. This confirms that the rate of information leakage is negligible in our scheme. Therefore, the proposed scheme successfully resists any kind of entropy attack.

Correlation coefficient means co-relation. It indicates the direction and degree (closeness) of linear relations between two variables X and Y . Correlation coefficient is denoted by ρ_{XY} or $\rho(X, Y)$, and is given by Equation (6).

$$\rho_{XY} = \rho(X, Y) = \frac{Cov(X, Y)}{\sqrt{Var(X)Var(Y)}} = \frac{Cov(X, Y)}{\sigma_X \sigma_Y} \quad (6)$$

where Cov and Var are variance and covariance. Cov and Var are given by Equation (7) and Equation (8) respectively.

$$\begin{aligned} Cov(X, Y) &= E[(X - EX)(Y - EY)] \\ &= E[XY] - (EX)(EY) \end{aligned} \quad (7)$$

where E is statistical expectation.

$$Var(X) = E^2(X) - E(X^2) \quad (8)$$

As we adopt RGB images in this paper, the two-dimensional correlation coefficient r is employed [20, 27] to compare between original and encrypted images, r is given by equation

$$r = \frac{\sum_{i=1}^M \sum_{j=1}^N (A_{[i][j]} - \bar{A})(B_{[i][j]} - \bar{B})}{\sqrt{\sum_{i=1}^M \sum_{j=1}^N (A_{[i][j]} - \bar{A})^2 \sum_{i=1}^M \sum_{j=1}^N (B_{[i][j]} - \bar{B})^2}} \quad (9)$$

Table 3: Experimental results of correlation coefficients analysis

Image	Image size	Variance	Standard Deviation
Zika virus	512×512	$1.69271186518e^{-36}$	$1.3010e^{-18}$
	256×256	0	0
MERS-CoV	512×512	$7.52316384526e^{-37}$	$8.6736e^{-19}$
	256×256	$4.70197740329e^{-38}$	$2.1684e^{-19}$
TBRF	512×512	$6.56270479213e^{-35}$	$3.3212e^{-16}$
	256×256	$2.32221569203e^{-37}$	$4.6241e^{-18}$

where A is original (plain) image Img_p , B is encrypted (scrambled) image Img_ϕ . $A_{[i][j]}$ and $B_{[i][j]}$ are the intensity of the pixel in i^{th} row and j^{th} column for A and B respectively, and \bar{A} is the mean of A and \bar{B} is the mean of B . The values of the correlation coefficient satisfy the relation $-1 \geq r \geq 1$. N and M are the total numbers of pixel in each column and row respectively.

Table 3 shows the variance and standard deviation of correlation coefficients analysis. As seen in this table, the coefficient correlation between neighboring pixels are very close to the ideal value 0. This indicates that there is significant differences between the original image and its corresponding encrypted image according to the pixel coordinates.

5.1 Histogram Analysis

An image histogram is a graphical representation that shows the distribution of the intensity of pixels in a digital image. Statistical attack or histogram analysis attack repeat a series of histogram analysis to deduce the secret key or plain-pixels. Therefore, encrypted image should have a histogram with a uniform distribution. Figures 5, 6 and 7 show the histogram of the selected images: “Zika virus”, “MERS-CoV” and “TBRF” respectively. Comparing the histograms of plain image with encrypted and decrypted images in each figure, it found that there is no resemblance between the histogram of original image and the histogram of encrypted image, while the histogram of original image is very similar to the histogram of decrypted image. Furthermore, the histograms of encrypted images are distributed uniformly. Hence, the proposed scheme is robust against histogram analysis attack.

6 Conclusion

In this paper, we have proposed a secure image encryption scheme for medical applications by incorporating the Arnold transformation and pairing-free identity-based authenticated key agreement protocol. After that, we have experimentally estimated the robustness and performance of our scheme. The analyses and results demonstrate that our scheme is efficient and secure. The long-term results of this effort is to offer a practical medical image water-

marking that provides authentication and integrity control.

References

- [1] D. Arroyo, C. Li, S. Li, G. Alvarez, and Wolfgang A. Halang, "Cryptanalysis of an image encryption scheme based on a new total shuffling algorithm," *Chaos, Solitons & Fractals*, vol. 41, no. 5, pp. 2613–2616, 2009.
- [2] M. Ashtiyani, P. M. Birgani, and H. M. Hosseini, "Chaos-based medical image encryption using symmetric cryptography," in *Information and Communication Technologies: From Theory to Applications, 2008. ICTTA 2008. 3rd International Conference on*, pp. 1–5, April 2008.
- [3] X. Cao, W. Kou, and X. Du, "A pairing-free identity-based authenticated key agreement protocol with minimal message exchanges," *Information Sciences*, vol. 180, no. 15, pp. 2895–2903, 2010.
- [4] G. Chen, Y. Mao, and C. K. Chui, "A symmetric image encryption scheme based on 3d chaotic cat maps," *Chaos, Solitons & Fractals*, vol. 21, no. 3, pp. 749–761, 2004.
- [5] J. Chen, J. Zhou, and K. W. Wong, "A modified chaos-based joint compression and encryption scheme," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 58, pp. 110–114, Feb 2011.
- [6] L. J. Chen and A. D. Shen, "A novel public key image cryptosystem based on elliptic curve and arnold cat map," in *Advanced Materials Research*, vol. 989, pp. 4183–4186. Trans Tech Publ, 2014.
- [7] W. B. Chen and X. Zhang, "Image encryption algorithm based on henon chaotic system," in *2009 International Conference on Image Analysis and Signal Processing*, pp. 94–97, April 2009.
- [8] R. Enayatifar, A. Abdullah, and I. F. Isnin, "Chaos-based image encryption using a hybrid genetic algorithm and a {DNA} sequence," *Optics and Lasers in Engineering*, vol. 56, pp. 83–93, 2014.
- [9] T. Gao and Z. Chen, "Image encryption based on a new total shuffling algorithm," *Chaos, Solitons & Fractals*, vol. 38, no. 1, pp. 213–220, 2008.
- [10] K. Hamdnaalla, A. Wahaballa, and O. Wahballa, "Digital image confidentiality depends upon arnold transformation and rc4 algorithms," *International Journal of Video & Image Processing and Network Security IJVIPNS-IJENS*, vol. 13, no. 04, pp. 6–17, 2013.
- [11] D. Hankerson, A. J. Menezes, and S. Vanstone, *Guide to elliptic curve cryptography*. Springer Science & Business Media, 2006.
- [12] X. Huang, "Image encryption algorithm using chaotic chebyshev generator," *Nonlinear Dynamics*, vol. 67, no. 4, pp. 2411–2417, 2011.
- [13] 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.
- [14] N. Koblitz, "Elliptic curve cryptosystems," *Mathematics of computation*, vol. 48, no. 177, pp. 203–209, 1987.
- [15] C. Li, S. Li, M. Asim, J. Nunez, G. Alvarez, and G. Chen, "On the security defects of an image encryption scheme," *Image Vision Comput.*, vol. 27, pp. 1371–1381, August 2009.
- [16] C. Li, Y. Liu, T. Xie, and Z. Q. Chen, "Breaking a novel image encryption scheme based on improved hyperchaotic sequences," *Nonlinear Dynamics*, vol. 73, no. 3, pp. 2083–2089, 2013.
- [17] 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.
- [18] S. V. Miller, "Use of elliptic curves in cryptography," in *Advances in Cryptology CRYPTO85 Proceedings*, pp. 417–426. Springer, 1985.
- [19] N. Mohananthini and G. Yamuna, "A study of dwt-svd based multiple watermarking scheme for medical images," *International Journal of Network Security*, vol. 17, no. 5, pp. 558–568, 2015.
- [20] A. M. Neto, A. C. Victorino, I. Fantoni, D. E. Zampieri, J. V. Ferreira, and D. A. Lima, "Image processing using pearson's correlation coefficient: Applications on autonomous robotics," in *Autonomous Robot Systems (Robotica), 2013 13th International Conference on*, pp. 1–6, April 2013.
- [21] N.K. Pareek, Vinod Patidar, and K.K. Sud, "Image encryption using chaotic logistic map," *Image and Vision Computing*, vol. 24, no. 9, pp. 926–934, 2006.
- [22] R. L. Rivest, A. Shamir, and L. Adleman, "A method for obtaining digital signatures and public-key cryptosystems," *Commun. ACM*, vol. 21, pp. 120–126, February 1978.
- [23] M. A. Scholl, M. K. Stine, J. Hash, P. Bowen, L. A. Johnson, C. D. Smith, and D. I. Steinberg. "Sp 800-66 rev. 1. an introductory resource guide for implementing the health insurance portability and accountability act (HIPAA) security rule,". tech. rep., Gaithersburg, MD, United States, 2008.
- [24] Ercan Solak and Cahit Okal, "Algebraic break of image ciphers based on discretized chaotic map lattices," *Information Sciences*, vol. 181, no. 1, pp. 227–233, 2011.
- [25] F. Y. Sun and Z. W. Lu, "Digital image encryption with chaotic map lattices," *Chinese Physics B*, vol. 20, no. 4, p. 040506, 2011.
- [26] X. Tong and M. Cui, "Image encryption scheme based on 3d baker with dynamical compound chaotic sequence cipher generator," *Signal Process.*, vol. 89, pp. 480–491, April 2009.

- [27] V. Tsagaris and V. Anastassopoulos, "Multispectral image fusion for improved rgb representation based on perceptual attributes," *International Journal of Remote Sensing*, vol. 26, no. 15, p. 3241C3254, 2005.
- [28] S. Voloshynovskiy, S. Pereira, T. Pun, J. J. Eggers, and J. K. Su, "Attacks on digital watermarks: Classification, estimation-based attacks, and benchmarks," *IEEE Communications Magazine*, vol. 39, pp. 118–126, 2001.
- [29] A. Wahaballa, H. Xiong, F. Li, Z. Qin, and Z. Qin, "Secure mobile agent-based english auction protocol using identity-based signature scheme," *Int. J. Security and Networks*, vol. 11, no. 4, pp. 175–187, 2016.
- [30] O. Wahballa, A. Wahaballa, F. Li, and C. Xu, "A secure and robust certificateless public key steganography based on svd-ddwt," *International Journal of Network Security*, vol. 18, no. 5, pp. 888–899, 2016.
- [31] X. Zhang, G. Zhu, W. Wang, M. Wang, and S. Ma, "Period law of discrete two-dimensional arnold transformation," in *Frontier of Computer Science and Technology (FCST), 2010 Fifth International Conference on*, pp. 565–569, Aug 2010.
- [32] G. Zhao, X. Yang, B. Zhou, and W. Wei, "RSA-based digital image encryption algorithm in wireless sensor networks," in *Signal Processing Systems (ICSPS), 2010 2nd International Conference on*, vol. 2, pp. V2–640–V2–643, July 2010.
- [33] C. Zhu, "A novel image encryption scheme based on improved hyperchaotic sequences," *Optics Communications*, vol. 285, no. 1, pp. 29–37, 2012.
- [34] Z. L. Zhu, W. Zhang, K. W. Wong, and H. Yu, "A chaos-based symmetric image encryption scheme using a bit-level permutation," *Information Sciences*, vol. 181, no. 6, pp. 1171–1186, 2011.

Abubaker Wahaballa is currently working as a Post-doctoral Fellow at School of Information and Software Engineering, University of Electronic Science and Technology of China UESTC. He received his PhD degree from UESTC in 2015. His current research interests include information security, cryptography, steganography, and DevOps.

Fagen Li Fagen Li received his Ph.D. degree in cryptography from Xidian University, Xian, China in 2007. He is now an associate professor in the School of Computer Science and Engineering, University of Electronic Science and Technology of China, Chengdu, China. His recent research interests include cryptography and network security. He has published more than 70 papers in the international journals and conferences.

Idris Ibn idris earned his BEng degree in Electrical Engineering at Ahmadu Bello University, Zaria, Nigeria in 2008 and MSc degree in Applied Instrumentation and Control at Glasgow Caledonian University, Scotland, United Kingdom in 2011. He is currently pursuing the PhD degree in Power Systems and Automation with the School of Electric Power, South China University of Technology, Guangzhou, China. His current research interests include Instruments Communication and Networking, Power Systems Operations and Control, and Information Security.

Chunxiang Xu received her B.Sc., M.Sc. and Ph.D. degrees at Xidian University, in 1985, 1988 and 2004 respectively, P.R. China. She is presently engaged in information security, cloud computing security and cryptography as a professor at University of Electronic Science Technology of China (UESTC).

Biography

Osman Wahballa received the B.S. degree in electrical engineering and computer engineering from Karary University, Department of Electrical Engineering in 2006, Khartoum, Sudan, and the M.S. degree in M.Sc. in Computer Engineering, Information Security form University of Electronic Science and Technology of China in 2013, Chengdu, China. He is currently working toward the Ph.D. degree in computer science from University of Electronic Science and Technology of China. His current research interests include information hiding, steganography, and cryptography.

An Enhanced Anonymous Password-based Authenticated Key Agreement Scheme with Formal Proof

Min Wu, Jianhua Chen, and Ruibing Wang

(Corresponding author: Min Wu)

School of Mathematic and Statistics, Wuhan University

Wuchang, Wuhan, Hubei 430072, China

(Email: wumin9246@163.com)

(Received May 23, 2016; revised and accepted Aug. 13 & Sept. 3, 2016)

Abstract

With the development of technology, the security of password-based authentication is becoming more and more significant. Recently, Lee et al. proposed an anonymous password-based authenticated key agreement scheme with non-tamper resistant smart card to reduce the computation cost of Wang et al.'s scheme. However, based on analysis, it shows that the scheme can't withstand smart card stolen or lost attack, user impersonation attack and server impersonation attack. Therefore, an enhanced scheme which can resist the attacks mentioned above is presented. By comparing the performance and security with other related schemes, our proposed scheme is more suitable for practical applications.

Keywords: Authentication Scheme; BNA Logic; Key Agreement; Network Security; Smart Card

1 Introduction

As the internet technology's development, password-based authentication with smart card is significant and widely used for remote system to access to computer network [1, 15]. To enhance the system security and management, research have been focused considerable attention on smart card based password authentication. Since Change and Wu [4] firstly proposed remote user authentication scheme using smart cards in 1993, many other password schemes were present [7, 12, 16, 18]. Traditionally, the smart card is assumed to be tamper-resistant. Namely, an adversary can't obtain the secret information about legal user stored in the smart card. However, recent research has been proved that the secret data stored in the smart card could be extracted by some means, such as monitoring the power consumption [2, 9, 14] or analyzing the leaked information [6, 13]. So such schemes based on the tamper resistance assumption of the smart card are

susceptible to various attacks like impersonation attacks, off-line password guessing attacks, etc.

In 2009, Kim and Chung [8] proposed a remote user authentication scheme which claimed that their scheme is secure. However in 2011, Li et al. [11] pointed out that Kim and Chung's scheme couldn't resist various attacks and further advanced a new remote authentication based on hash function. In their scheme, they suggested that their scheme not only remedy the flaws of Kim and Chung's scheme, but also secure. But in 2012, Wang et al. [17] demonstrated that Li et al.'s scheme is insecure against denial of service attack and off-line password guessing attack under the non-tamper resistance assumption of the smart card. Moreover, their scheme failed to provide user anonymity and forward secrecy. In order to solve the problems mentioned above, Wang et al. presented a robust authentication scheme based on the secure one-way hash function and the well-known discrete logarithm problem. Later, Lee et al. [10] putted forward that Wang et al.'s scheme had high computational overhead. In order to reduce the overhead, they proposed an anonymous authentication scheme with non-tamper resistant smart cards based on password, and proved that their scheme meets all the criteria required for the authenticated key agreement scheme and eliminates security threats. Nevertheless, it indicated that their scheme is prone to smart card stolen or lost attack, user impersonation attack and server impersonation attack base on our analysis. In additional, their scheme can't provide mutual authentication. Then, an enhanced key agreement scheme with non-tamper resistant smart cards is presented. The remainder of the article is sketched as follows. In Section 2, we briefly review Lee et al.'s scheme. Section 3 presents the security analysis of Lee et al.'s scheme. In Section 4, we present an enhanced scheme. The security analysis of the proposed scheme is given in Section 5, and efficiency comparison between our scheme and other related ones is showed in Section 6. Ultimately, in Section 7,

we reach the conclusion.

2 Review of Lee et al.'s Scheme

In this section, we will briefly review of Lee et al.'s scheme, which comprises four phases: registration phase, login phase, authentication phase and password change phase. The notations used in this article are described in Table 1.

Table 1: Notation

Notation	Description
U_i/U_k	user i/k
S_i	server i
E	attacker
PW_i	U_i 's password
ID_i	U_i 's identity
x	secret key generated by S_i
y	public key generated by S_i
b	a random number generated by U_i
v	a random number generated by U_i
w	a random number generated by S_i
$h(\cdot)$	a one-way hash function
\parallel	concatenation
\oplus	bitwise exclusive-or operation

2.1 Registration Phase

S_i generates x as the server's private key which is only kept secret by himself/herself, and computes $y = g^x \text{ mod } n$ as its corresponding public key which is stored inside each user's smart card. If a user U_i wishes to be a legal user of the system so that he/she can utilize resources provided by the server, U_i should execute the following steps.

- U_i first selects his/her identity ID_i and password PW_i . Then, U_i generates a random number b , computes $h(b\parallel PW_i)$ and sends $\{ID_i, h(b\parallel PW_i)\}$ to S_i .
- S_i checks the validity of ID_i . If it is validity, S_i calculates

$$\begin{aligned}
 C_1 &= h(h(ID_i) \oplus x), \\
 C_2 &= C_1 \oplus h(b\parallel PW_i) \oplus h(ID_i), \\
 C_3 &= h(C_1), \\
 C_4 &= h(b\parallel PW_i) \oplus h(x\parallel y).
 \end{aligned}$$

Then S_i issues a smart card including $\{C_2, C_3, C_4, h(\cdot), n, g, y\}$ to U_i via a secure channel.

- U_i computes $B = b \oplus ID_i \oplus PW_i$, and stores B in the smart card.

2.2 Login Phase

When U_i logs in the system, he/she can perform the next steps.

- U_i inserts his/her smart card into a card reader and enters the identity ID_i , password PW_i . The smart card SC computes $b' = B \oplus ID_i \oplus PW_i$, $C'_1 = C_2 \oplus h(b' \parallel PW_i) \oplus h(ID_i)$, $C'_3 = h(C'_1)$, and compares C'_3 with C_3 stored in the smart card. Only if the equation holds, SC performs the following steps.
- SC generates a random number v and computes $V = g^v \text{ mod } n$, $h(x\parallel y) = c_4 \oplus h(b\parallel PW_i)$, $CID_i = h(ID_i) \oplus h(V\parallel h(x\parallel y))$, $M_1 = h(CID_i\parallel V\parallel C_1)$. Then, U_i sends login request message $\{CID_i, V, M_1\}$ to S_i .

2.3 Authentication Phase

U_i and S_i achieve mutual authentication as follows.

- Upon receiving the login message $\{CID_i, V, M_1\}$, S_i computes $h(x\parallel y)$, $h(ID_i) = CID_i \oplus h(V\parallel h(x\parallel y))$, $C'_1 = h(h(ID_i) \oplus x)$, $M'_1 = h(CID_i\parallel V\parallel C'_1)$, and checks whether M'_1 equals to the received M_1 . If they are not equal, the session is terminated. Otherwise, S_i selects a random number w and computes $W = g^w \text{ mod } n$, $SK = V^w \text{ mod } n$, $M_2 = h(SK\parallel W\parallel C'_1)$. Then, S_i sends $\{M_2, W\}$ to U_i .
- SC receives the message and computes the session key $SK' = W^v \text{ mod } n$. And, SC verifies M_2 with the computed value of $h(SK'\parallel W\parallel C_1)$. If the verification holds, SC computes $M_3 = h(M_2\parallel C_1\parallel SK')$ and send $\{M_3\}$ to S_i .
- Upon receiving $\{M_3\}$, S_i computes $M'_3 = h(M_2\parallel C_1\parallel SK')$ and checks whether the equation $M'_3 = M_3$ holds. If it holds, S_i and U_i finish mutual authentication, and share a common session key $SK = g^{vw} \text{ mod } n$. Otherwise, the session is terminated.

2.4 Password Change Phase

Assume that SC has the ability to detect the login failure trials. If the failure times exceed a given number, SC will be soon locked to prevent from guessing password attack.

- U_i inserts the smart card into a card reader and inputs identity ID_i , password PW_i and a new password PW_i^{new} .
- SC calculates $b' = B \oplus ID_i \oplus PW_i$, $C'_1 = C_2 \oplus h(b' \parallel PW_i \oplus h(ID_i))$, $C'_3 = h(C'_1)$ and verifies whether $C'_3 = C_3$. If they are the same, SC accepts the change request. Otherwise, the session is terminated.
- SC computes $B^{new} = b \oplus ID_i \oplus PW_i^{new}$, $C_2^{new} = C'_1 \oplus h(b' \parallel h(b' \parallel PW_i^{new})) \oplus h(ID_i)$, $C_4^{new} = C_4 \oplus h(b' \parallel PW_i) \oplus h(b' \parallel PW_i^{new})$. Finally, SC replace C_2, C_4, B with $C_2^{new}, C_4^{new}, B^{new}$ in the smart card.

3 Security Analysis of Lee et al.'s Scheme

In Lee et al.'s scheme, they claim that their scheme can resist some attacks, containing off-line password guessing attack, user impersonation attack, server masquerading attack, and so on. By analysis and study, we find that the scheme fails to resist the attacks mentioned above. The details are as follows.

3.1 Smart Card Stolen or Loss Attack

Assume that U_i 's smart card was stolen by a legal but malicious user U_k , and U_k had monitored the login request message $\{CID_i, V, M_1\}$ which was sent to S by U_i .

A legal but malicious user U_k acquires $\{C_2^*, C_3^*, C_4^*, h(\cdot), n, g, B^*\}$ from his/her own smart card and computes $b^* = B^* \oplus ID_k \oplus PW_k$, $h(x||y) = C_4^* \oplus h(b^*||PW_k)$. And the value of $h(x||y)$ is not changed for every user. Then U_k can obtain $h(ID_i)$ and C_1 by computing $h(ID_i) = CID_i \oplus h(V||h(x||y))$, $C_1 = C_2 \oplus h(b||PW_i) \oplus h(ID_i) = C_2 \oplus C_4 \oplus h(x||y) \oplus h(ID_i)$ where C_2, C_4 is extracted from U_i 's smart card. Then, U_k can continue guesses the identity as follows.

- 1) Guess an identity ID_i' .
- 2) compute $h(ID_i')$ and compare it with the values of $CID_i \oplus h(V||h(x||y))$. If they are not equal, go back to 1). Otherwise, U_k finds the user U_i 's identity ID_i .

After acquiring the user U_i 's identity ID_i , U_k can go on continuing guess user's password.

- 1) Guess a password PW_i' .
- 2) Compute $b' = B \oplus ID_i \oplus PW_i'$, $C_4' = h(b'||PW_i') \oplus h(x||y)$, where B is extracted from U_i 's smart card and $h(x||y)$ can be obtained by Step 1. Then U_k verifies $C_4' \stackrel{?}{=} C_4$. If it holds, U_k finds the correct password PW_i .

3.2 User Impersonation Attack

From Section 3.1, we know that a legal but malicious user U_k can obtain $h(x||y)$, $h(ID_i)$, C_1 . Then he/she can forge the login request message $\{CID_i, V, M_1\}$ to disguise the user U_i .

- 1) U_k generates a random number v^* and computes $V^* = g^{v^*} \bmod n$, $CID_i^* = h(ID_i) \oplus h(V^*||h(x||y))$, $M_1^* = h(CID_i^* || V^* || C_1)$. Then, U_k sends $\{CID_i^*, V^*, M_1^*\}$ to S_i .
- 2) S_i computes $h(x||y)$, $h(ID_i) = CID_i^* \oplus h(V^*||h(x||y))$, $C_1' = h(h(ID_i) \oplus x)$, $M_1' = h(CID_i^*||V^*||C_1')$, and checks whether M_1' equals to the received M_1^* . If they are equal, then S_i selects a random number w^* and computes $W^* = g^{w^*} \bmod n$, $SK^* = (V^*)^{w^*} \bmod n$,

$M_2^* = h(SK^*||W^*||C_1')$. Then, S_i sends $\{M_2^*, W^*\}$ to U_i .

- 3) U_k computes the session key $SK' = (W^*)^{v^*} \bmod n$, $M_3^* = h(M_2^*||C_1||SK^*)$ and send $\{M_3^*\}$ to S_i .
- 4) S_i computes $M_3' = h(M_2^*||C_1'||SK^*)$ and checks whether the equation $M_3 = M_3^*$ holds. As $SK^* = (V^*)^{w^*} \bmod n = (g^{v^*})^{w^*} \bmod n = (g^{w^*})^{v^*} \bmod n = (W^*)^{v^*} \bmod n = SK$, $M_3' = h(M_2^*||C_1'||SK^*) = h(M_2^*||C_1||SK') = M_3^*$. S_i authenticates U_k as U_i .

3.3 Server Impersonation Attack

A legal but malicious user U_k acquires $h(x||y), h(ID_i), C_1$ by the method mentioned in Section 3.1, then U_k can impersonate server S_i to communicate with U_i .

- 1) When U_i sends the login request message $\{CID_i, V, M_1\}$ to S_i , U_k eavesdrops the message, selects a random number w^* and computes $W^* = g^{w^*} \bmod n$, $SK^* = V^{w^*} \bmod n$, $M_2^* = h(SK^*||W^*||C_1)$. Then, S_i sends $\{M_2^*, W^*\}$ to U_i .
- 2) When U_i receives the message, the smart card computes the session key $SK' = (W^*)^v \bmod n$. $SK' = (W^*)^v \bmod n = (g^{w^*})^v \bmod n = (g^v)^{w^*} \bmod n = (V)^{w^*} \bmod n = SK^*$, so $M_2^* = h(SK'||W^*||C_1)$. Then, SC computes $M_3 = h(M_2||C_1||SK')$ and send $\{M_3\}$ to S_i .

Thus, U_k is authenticated as the legitimate server by the user U_i .

4 Our Proposed Scheme

In this section, we propose a new scheme based on Lee et al.'s scheme, which can resist the attacks mentioned in Section 3. It composes four phase: registration phase, login phase, authentication phase and password change phase. The detail description of each phase are shown below.

4.1 Registration Phase

S_i generates x as the server's private key which is only kept secret by himself/herself, and computes $y = g^x \bmod n$ as its corresponding public key which is stored inside each user's smart card. A user U_i must register to be a legal user of the system, before utilizing resources provided by the server.

- U_i first selects his/her identity ID_i and password PW_i . Then, U_i generates a random number b , computes $RPW_i = h(b||PW_i)$ and sends $\{ID_i, RPW_i\}$ to S_i .

Table 2: The proposed scheme of registration phase

U_i	S_i
$RPW_i = h(b PW_i)$	ID_i, RPW_i
$B = b \oplus ID_i \oplus PW_i$ stores B in the smart card	Server's public key $y = g^x \text{mod} n$. checks the validity of ID_i generates a random number d $C_1 = h(ID_i x), C_2 = C_1 \oplus RPW_i$ $C_3 = h(C_1 d), C_4 = h(C_1 RPW_i) \oplus d,$ $D = g^d \text{mod } n, C_5 = h(C_1 \oplus ID_i) \oplus h(x y D).$
	$\xleftarrow{\text{smart card}}$

- S_i checks the validity of ID_i . If it is validity, S_i generates a random number d for user U_i . Then S_i performs the following computations. $C_1 = h(ID_i||x)$, $C_2 = C_1 \oplus RPW_i$, $C_3 = h(C_1||d)$, $C_4 = h(C_1||RPW_i) \oplus d$, $D = g^d \text{mod } n$, $C_5 = h(C_1 \oplus ID_i) \oplus h(x||y||D)$. Then S_i sends a smart card including $\{C_2, C_3, C_4, C_5, h(\cdot), n, g, y\}$ to U_i via a secure channel.
- U_i computes $B = b \oplus ID_i \oplus PW_i$, and stores B in the smart card.

4.2 Login Phase

When U_i logs in the system, he/she can perform the next steps.

- U_i inserts his/her smart card into a card reader and enters the identity ID_i , password PW_i . The smart card SC computes $b = B \oplus ID_i \oplus PW_i$, $RPW_i = h(b||PW_i)$, $C_1 = C_2 \oplus RPW_i$, $d = C_4 \oplus h(C_1||RPW_i)$, $C_3 = h(C_1||d)$, and compares C_3 with C_3 stored in the smart card. Only if the equation holds, SC performs the following steps.
- SC generates a random number v and computes $V = g^v \text{mod } n$, $D = g^d \text{mod } n$, $h(x||y||D) = C_5 \oplus h(C_1||ID_i)$, $CID_i = ID_i \oplus h(V||h(x||y||D))$, $F_1 = RPW_i \oplus h(C_1||ID_i)$, $F_2 = C_4 \oplus h(V||C_1) \oplus h(x||y||D)$, $M_1 = h(ID_i||RPW_i||V||C_1||d)$. Then, U_i sends login request message $\{CID_i, V, D, F_1, F_2, M_1\}$ to S_i .

4.3 Authentication Phase

U_i and S_i achieve mutual authentication as follows.

- Upon receiving the login message $\{CID_i, V, D, F_1, F_2, M_1\}$, S_i computes $h(x||y||D)$, $ID_i = CID_i \oplus h(V||h(x||y||D))$, $C_1 = h(ID_i||x)$, $RPW_i = F_1 \oplus h(C_1||ID_i)$, $C_4 = F_2 \oplus h(V||C_1) \oplus h(x||y||D)$, $d = C_4 \oplus h(C_1 \oplus RPW_i)$, $M_1^* = h(ID_i||RPW_i||V||C_1||d)$, and checks whether M_1^* equals to the received M_1 . If they are not equal, the session is terminated. Otherwise, S_i selects a random number w and computes $W = g^w \text{mod } n$, $SK = V^w \text{mod } n$, $M_2 =$

$h(SK||W||C_1||RPW_i||d)$. Then, S_i sends $\{M_2, W\}$ to U_i .

- SC receives the message and computes the session key $SK' = W^v \text{mod } n$. And, SC verifies M_2 with the computed value of $h(SK'||W||C_1||RPW_i||d)$. If the verification holds, SC computes $M_3 = h(M_2||C_1||SK'||d)$ and send $\{M_3\}$ to S_i .
- Upon receiving $\{M_3\}$, S_i computes $M_3^* = h(M_2||C_1||SK||d)$ and checks whether the equation $M_3^* = M_3$ holds. If it holds, S_i and U_i finish mutual authentication, and share a common session key $SK = g^{vw} \text{mod } n$. Otherwise, the session is terminated.

4.4 Password Change Phase

Assume that SC has the ability to detect the login failure trials. If the failure times exceed a given number, SC will be soon locked to prevent from guessing password attack.

- U_i inserts the smart card into a card reader and inputs identity ID_i , password PW_i and a new password PW_i^{new} .
- SC calculates $b = B \oplus ID_i \oplus PW_i$, $RPW_i = h(b||PW_i)$, $C_1 = C_2 \oplus RPW_i$, $d = C_4 \oplus h(C_1||RPW_i)$, $C_3 = h(C_1||d)$ and verifies whether $C_3 = C_3$. If they are the same, SC accepts the request. Otherwise, the session is terminated.
- SC computes $B^{new} = b \oplus ID_i \oplus PW_i^{new}$, $RPW_i^{new} = h(b||PW_i^{new})$, $C_2^{new} = C_1 \oplus RPW_i^{new}$, $C_4^{new} = d \oplus h(C_1||RPW_i^{new})$. Finally, SC replace C_2, C_4, B with $C_2^{new}, C_4^{new}, B^{new}$ in the smart card.

5 Security Analysis

The proposed scheme advanced Lee et als scheme and can resist the attacks analyzed above. The details are described in the following content.

Table 3: The proposed scheme of the login and authentication phase

U_i	S_i
inputs ID_i, PW_i computes $b = B \oplus ID_i \oplus PW_i$, $RPW_i = h(b PW_i), C_1 = C_2 \oplus RPW_i$, $d = C_4 \oplus h(C_1 RPW_i)$, $C'_3 = h(C_1 d)$, verifies $C'_3 \stackrel{?}{=} C_3$. selects a random number v , computes $V = g^v \text{mod} n$, $D = g^d \text{mod} n$, $h(x y D) = C_5 \oplus h(C_1 \oplus ID_i)$, $CID_i = ID_i \oplus h(V h(x y D))$, $F_1 = RPW_i \oplus h(C_1 ID_i)$, $F_2 = C_4 \oplus h(V C_1) \oplus h(x y D)$ $M_1 = h(ID_i RPW_i V C_1 d)$	$CID_i, V, D, F_1, F_2, M_1 \xrightarrow{\hspace{2cm}}$
$SK' = W^v \text{ mod } n$, verifies $M_2 \stackrel{?}{=} h(SK' W C_1 RPW_i d)$, computes $M_3 = h(M_2 C_1 SK' d)$.	$\xleftarrow{M_2, W}$ computes $h(x y D)$, $ID_i = CID_i \oplus h(V h(x y D))$, $C_1 = h(ID_i x)$, $RPW_i = F_1 \oplus h(C_1 ID_i)$, $C_4 = F_2 \oplus h(V C_1) \oplus h(x y D)$, $d = C_4 \oplus h(C_1 \oplus RPW_i)$, $M_1^* = h(ID_i RPW_i V C_1 d)$, checks $M_1^* \stackrel{?}{=} M_1$. selects a random number w , computes $W = g^w \text{ mod } n$, $SK = V^w \text{ mod } n$, $M_2 = h(SK W C_1 RPW_i d)$.
	$\xrightarrow{M_3}$ computes $M_3^* = h(M_2 C_1 SK d)$ checks $M_3^* \stackrel{?}{=} M_3$

5.1 Analysis the Proposed Scheme with BNA Logic

We analyzes out proposed scheme with BNA logic [3] in this section. The main notations of the BNA logic are shown in Table 4. Note that symbols P and Q stands for principals, X and Y range over statement, and K represent encryption keys.

Table 4: Notations of BNA logic

Notation	Meaning
$P \models X$	P believes that X is true.
$P \triangleleft X$	P once received a message including X .
$P \sim X$	P once said X .
$P \Rightarrow X$	P has jurisdiction over X .
$\#(X)$	X is fresh.
$(X, Y)_K$	X and Y are hashed with the key K .
$\{X, Y\}_K$	X and Y are encrypted with the key K .
$P \xleftrightarrow{K} Q$	P communicates with Q by a shared key K .

1) Idealization forms

$$\begin{aligned}
 U_i: & (ID_i, V)_{U_i \xleftrightarrow{h(x||y||D)} S_i}, & V, & & U_i \xleftrightarrow{d} S_i, \\
 & (RPW_i, ID_i)_{U_i \xleftrightarrow{h(ID_i||x)} S_i}, \\
 & (C_4, V, U_i \xleftrightarrow{h(x||y||D)} S_i)_{U_i \xleftrightarrow{h(ID_i||x)} S_i},
 \end{aligned}$$

$$\begin{aligned}
 & (ID_i, RPW_i, V, U_i \xleftrightarrow{d} S_i)_{U_i \xleftrightarrow{h(ID_i||x)} S_i}, \\
 & ((U_i \xleftrightarrow{SK} S_i, W, RPW_i, U_i \xleftrightarrow{d} S_i)_{U_i \xleftrightarrow{h(ID_i||x)} S_i}, U_i \xleftrightarrow{SK} S_i, U_i \xleftrightarrow{d} S_i)_{U_i \xleftrightarrow{h(ID_i||x)} S_i} \\
 S_i: & (U_i \xleftrightarrow{SK} S_i, W, RPW_i, U_i \xleftrightarrow{d} S_i)_{U_i \xleftrightarrow{h(ID_i||x)} S_i}, W
 \end{aligned}$$

2) Security goals

- G1 $S_i \models U_i \models U_i \xleftrightarrow{SK} S_i$
- G2 $S_i \models U_i \xleftrightarrow{SK} S_i$
- G3 $U_i \models S_i \models U_i \xleftrightarrow{SK} S_i$
- G4 $U_i \models U_i \xleftrightarrow{SK} S_i$

3) Initiative assumption

- A1 $U_i \models U_i \xleftrightarrow{h(ID_i||x)} S_i$
- A2 $S_i \models U_i \xleftrightarrow{h(ID_i||x)} S_i$
- A3 $U_i \models U_i \xleftrightarrow{d} S_i$
- A4 $S_i \models U_i \xleftrightarrow{d} S_i$
- A5 $U_i \models U_i \xleftrightarrow{h(x||y||D)} S_i$
- A6 $S_i \models U_i \xleftrightarrow{h(x||y||D)} S_i$
- A7 $S_i \models U_i \Rightarrow U_i \xleftrightarrow{SK} S_i$
- A8 $U_i \models S_i \Rightarrow U_i \xleftrightarrow{SK} S_i$

Table 5: BNA logical postulates

Rule	Formula	Meaning
Message-meaning rule	$\frac{P \equiv P \xleftarrow{K} Q, P \triangleleft \{X\}_K}{P \equiv Q \sim X}$	If P believes that K is the secret key shared by P with Q , and P sees X encrypted with K , then P believes that Q once said X .
Nonce-verification rule	$\frac{P \equiv \#(X), P \equiv Q \sim X}{P \equiv Q \equiv X}$	If P believes that X is fresh and Q once said X , then P believes that Q believes X .
Freshness-conjunction rule	$\frac{P \equiv \#(X)}{P \equiv \#(X, Y)}$	If P believes that X is fresh, then P believes that (X, Y) is fresh.
Jurisdiction rule	$\frac{P \equiv Q \Rightarrow X, P \equiv Q \equiv X}{P \equiv X}$	If P believes that Q controls X and P believes Q believes X , then P believes X .

4) Scheme analysis

The main analysis of our proposed scheme is described as follows: Since $S_i \triangleleft ((U_i \xleftarrow{SK} S_i, W, RPW_i, U_i \xleftrightarrow{d} S_i)_{U_i \xleftarrow{h(ID_i||x)} S_i}, U_i \xleftarrow{SK} S_i, U_i \xleftrightarrow{d} S_i)_{U_i \xleftarrow{h(ID_i||x)} S_i}$ and $S_i |\equiv U_i \xleftarrow{h(ID_i||x)} S_i$, we can know

$$S_i |\equiv U_i |\sim ((U_i \xleftarrow{SK} S_i, W, RPW_i, U_i \xleftrightarrow{d} S_i)_{U_i \xleftarrow{h(ID_i||x)} S_i}, U_i \xleftarrow{SK} S_i, U_i \xleftrightarrow{d} S_i) \quad (1)$$

based on message-meaning rule.

According to freshness-conjunction rule and $S_i |\equiv \#(W)$, we can derive

$$S_i |\equiv \#((U_i \xleftarrow{SK} S_i), W, RPW_i, U_i \xleftrightarrow{d} S_i)_{U_i \xleftarrow{h(ID_i||x)} S_i}, U_i \xleftarrow{SK} S_i, U_i \xleftrightarrow{d} S_i). \quad (2)$$

On the basis of Equations (1), (2) and nonce-verification rule, the following can be derived

$$S_i |\equiv U_i |\equiv ((U_i \xleftarrow{SK} S_i, W, RPW_i, U_i \xleftrightarrow{d} S_i)_{U_i \xleftarrow{h(ID_i||x)} S_i}, U_i \xleftarrow{SK} S_i, U_i \xleftrightarrow{d} S_i). \quad (3)$$

The G1 $S_i |\equiv U_i |\equiv U_i \xleftarrow{SK} S_i$ will be deduced from Equation (3).

Based on A7, G1 and jurisdiction rule, we can derive G2 $S_i |\equiv U_i \xleftarrow{SK} S_i$.

Since $U_i \triangleleft (U_i \xleftarrow{SK} S_i, W, RPW_i, U_i \xleftrightarrow{d} S_i)_{U_i \xleftarrow{h(ID_i||x)} S_i}$ and $U_i |\equiv U_i \xleftarrow{h(ID_i||x)} S_i$, we can know

$$U_i |\equiv S_i |\sim (U_i \xleftarrow{SK} S_i, W, RPW_i, U_i \xleftrightarrow{d} S_i) \quad (4)$$

based on message-meaning rule.

If $M_3 = h(M_2||C_1||SK'd)$, $U_i |\equiv \#(W)$. According to freshness-conjunction rule, we can derive

$$U_i |\equiv \#(U_i \xleftarrow{SK} S_i, W, RPW_i, U_i \xleftrightarrow{d} S_i). \quad (5)$$

On the basis of Equations (4), (5) and nonce-verification rule, the following can be derived

$$U_i |\equiv S_i |\equiv (U_i \xleftarrow{SK} S_i, W, RPW_i, U_i \xleftrightarrow{d} S_i). \quad (6)$$

The G3 $U_i |\equiv S_i |\equiv U_i \xleftarrow{SK} S_i$ will be deduced from Equation (6).

Based on A8, G3 and jurisdiction rule, we can derive G4 $U_i |\equiv U_i \xleftarrow{SK} S_i$.

5.2 Informal Security Analysis

5.2.1 User Anonymity

- 1) A legal but malicious user U_k acquires $\{C_2^*, C_3^*, C_4^*, C_5^*, h(\cdot), n, g, y, B^*\}$ from his/her own smart card and computes $b^* = B^* \oplus ID_k \oplus PW_k$, $RPW_k = h(b^*||PW_k)$, $C_1^* = C_2^* \oplus RPW_k$, $d^* = C_4^* \oplus h(C_1^*||RPW_k)$, $D^* = g^{d^*} \bmod n$, $h(x||y||D^*) = C_5^* \oplus h(C_1 \oplus ID_k)$. U_k can't obtain any common values for every legal user.
- 2) Even If U_k obtain $\{C_2, C_3, C_4, C_5, h(\cdot), n, g, Y, B\}$ from U_i 's smart card, he/she impossible to get d without knowing C_1 , RPW_i , or $h(x||y||D)$ without the values of C_1 , ID_i .
- 3) In unsecure channels, U_k intercepts the message $\{CID_i, V, D, F_1, F_2, M_1\}$, and tries to trace the user U_i . But the user U_i communicates with S_i by CID_i instead of his/ her own identity ID_i . It is infeasible to derive ID_i without knowing $h(x||y||D)$. On the other hand, it is hard to get the random number d from $D = g^d \bmod n$ due to discrete logarithm problem.

Consequently, any legal but malicious user cannot obtain some useful values concerning with user U_i .

5.2.2 Offline Password Guessing Attack

- 1) Form the analysis of Section 5.2.1, we know that any legal but malicious user U_k cannot the common value $h(x||y)$ for all legal users.
- 2) If U_k acquires $\{C_2, C_3, C_4, C_5, h(\cdot), n, g, y, B\}$ from U_i 's smart card, he/she has to guess the user U_i 's identity ID_i and password PW_i correctly at the same time to compute $b = B \oplus ID_k \oplus PW_k$. As we all known, it is difficult to guess the two parameters chosen freely by the user at the same time in polynomial time. And the proposed scheme can provide user anonymity by the above analysis. Furthermore, the adversary needs to know the server's private key x to compute $C_1 = h(ID_i||x)$, $RPW_i = h(b||PW_i)$. Then, he/she could get right password by comparing $C_1 \oplus RPW_i$ with C_2 .
- 3) Assume U_k intercepts the message $\{CID_i, V, D, F_1, F_2, M_1\}$ which U_i once sent to S_i . However U_k does not have the knowledge of b , C_1 and ID_i , the verification of the computed $F_1 = RPW_i \oplus h(C_1||ID_i)$ will fail.

5.2.3 Stolen Verifier Attack

The server S_i does not store any sensitive verification information corresponding to users in its database in our proposed scheme. Therefore even if any adversary accesses the server's database, he/she is impossible to gain any verification information related to registered users. So, the proposed scheme can withstand stolen verifier attack.

5.2.4 Insider Attack

Assume that the privileged user gets ID_i , RPW_i when a legal user U_i registers to the system S_i . However, the privileged couldn't extract PW_i from RPW_i due to one-way property of hash function. At the same time, PW_i is protected by random number b , and the privileged user is not able to guess the right password. Thus, the proposed scheme can resist insider attack.

5.2.5 Replay Attack

Suppose that an adversary E eavesdrops the login request message and tries to perform replay attack in future. Upon receiving $\{CID_i, V, D, F_1, F_2, M_1\}$ from E , the server S_i verifies $M_1 \stackrel{?}{=} h(ID_i||RPW_i||V||C_1||d)$. The message has not been changed by E , so S_i selects a random number w^* and computes $W^* = g^{w^*} \bmod n$, $SK^* = V^{w^*} \bmod n$, $M_2^* = h(SK^*||W^*||C_1||RPW_i||d)$. Then, S_i sends $\{M_2^*, W^*\}$ to the adversary E . It is indispensable for the adversary E to reply $\{M_3\}$ to S_i , where $M_3 = h(M_2||C_1||SK||d)$. Because E not only couldn't compute SK without random number v , but also couldn't get C_1 and d . Thus, the server cannot authenticate E . Namely, the scheme is secure against replay attack.

5.2.6 User Impersonation Attack

If an adversary E wants to pretend U_i to communicate with S_i , he/she must forge the login request message $\{CID_i, V, D, F_1, F_2, M_1\}$. Then, he/she selects a random number v^* and computes $V^* = g^{v^*} \bmod n$, $U^* = Y^{v^*}$. Unfortunately, E couldn't compute CID_i^* without the user U_i 's identity ID_i , server's private key x . Meanwhile, U_k requires to compute $F_1^* = RPW_i \oplus h(C_1||ID_i)$, $F_2^* = C_4 \oplus h(V^*||C_1) \oplus h(x||y||D)$, which is not possible, since E does not know ID_i , RPW_i , x . That is, the proposed scheme is able to against the user spoofing attack.

5.2.7 Server Impersonation Attack

If an adversary E eavesdrops the login request message $\{CID_i, V, F_3, F_4, M_1\}$ from user U_i , he/she performs the following steps to act as the legal server S_i . E must compute $M_2 = h(SK||W||C_1||RPW_i||d)$ to respond the login request message. Even If U_k selects a random number w^* and computes $W^* = g^{w^*}$, $SK^* = V^{w^*} \bmod n$, he/she cannot forge M_2 without RPW_i , C_1 , d . From the above analysis, our proposed scheme could resist server impersonation attack.

5.2.8 Mutual Authentication

- 1) In the proposed scheme, S_i authenticates U_i by checking the validity of equation $M_3 \stackrel{?}{=} h(M_2||C_1||SK||d)$. We have demonstrated that the proposed scheme can provide user anonymity and off-line password guessing attack. If an adversary replays the former login request message $\{CID_i, V, D, F_1, F_2, M_1\}$ sent to S_i by U_i , he/she would fail according to the analysis of section 5.2.5. On the other hand, suppose the adversary forge the login request message to cheat the server, we will find that it is impossible by the analysis of Section 5.2.6.
- 2) On the contrary, the legal user U_i authenticates S_i by comparing M_2 with the computed value $h(SK||W||C_1||RPW_i||d)$. Based on the analysis of Section 5.2.7, no one can act as legal user to deceive the server.

Therefore, the proposed scheme can provide mutual authentication.

5.2.9 Forward Secrecy

In the improved scheme, the user U_i and the server S_i establish the same session key $SK = W^v \bmod n = V^w \bmod n = g^{vw} \bmod n$. Due to discrete logarithm problem (DLP), no one is able to compute the previously established session keys without knowing v , w . As a result, the proposed scheme provides perfect forward secrecy.

Table 6: Performance comparison

	Total of login and authentication phase	Time
[17]	$6T_e + 11T_h$	13.7303
[10]	$4T_e + 13T_h$	9.1575
[5]	$6T_e + 5T_h$	13.7261
[20]	$8T_e + 7T_h$	18.3017
[21]	$6T_e + 8T_h$	13.7282
Ours	$5T_e + 17T_h$	11.4474

6 Performance Analysis

In this section, we will show efficiency and functionality comparison among our proposed scheme and other related schemes. According to Wu et al.'s report [19], the time of executing one modular exponentiation is 2.2871ms, while the computation time of a one-way hash function is 0.0007ms. For the convenience, we define the following notations used in this section.

- T_h : time for executing a one-way hash function.
- T_e : time for executing exponential operation.
- T_{\oplus} : time for executing XOR operation.

Compared with T_e and T_h , the time of executing XOR operation can be neglected. Usually, a legal user only needs to perform once registration operation, but login and authentication phase are carried out more times in a short time. So we display the comparison of the computational cost in login and authentication phase among these schemes in Table 6. In Table 7, we show security comparison between our proposed scheme and other related ones.

From the comparison of Table 6 and Table 7, we can conclude that the performance of our scheme has better efficiency than other related schemes. Taking all into account, the proposed scheme is more suitable for practical applications.

7 Conclusions

In this paper, we review an anonymous password-based authenticated key agreement scheme with non-tamper resistant smart cards which is proposed by Lee et al. to reduce time cost under the condition of safety. However, Lee et al.'s scheme is vulnerable to smart card stolen or lost attack, user impersonation attack, server impersonation attack and cannot provide mutual authentication. To overcome the weakness mentioned above, an improved scheme is proposed. Finally, we demonstrate that our scheme is more secure and applicable to practice by comparing the performance and efficiency of our scheme with other related ones.

References

- [1] N. Anwar, I. Riadi, A. Luthfi, "Forensic SIM card cloning using authentication algorithm," *International Journal of Electronics and Information Engineering*, vol. 4, no. 2, pp. 71–81, 2016.
- [2] A. Bogdanov, I. Kizhvatov, "Beyond the limits of dpa: Combined side-channel collision attacks," *IEEE Transactions on Computers*, vol. 61, no. 8, pp. 1153–1164, 2012.
- [3] M. Burrows, M. Abadi, R. Needham, "A logic of authentication," *ACM Transaction on Computer System*, vol. 8, no. 1, pp. 18–36, 1990.
- [4] C. C. Chang, T. C. Wu, "Remote password authentication with smart cards," *IEE Proceedings-E*, vol. 138, no. 3, pp. 165–168, 1993.
- [5] Y. Chen, J. S. Chou, C. H. Huang, "Improvements on two password-based authentication protocols," *Cryptology ePrint Archive*, Report 2009/561, Nov. 1990.
- [6] T. Kasper, D. Oswald, C. Paar, "Side-channel analysis of cryptographic RFIDs with analog demodulation," in *7th International Workshop on RFID Security and Privacy (RFIDsec'11)*, pp. 61–77, 2011.
- [7] M. Khan, S. Kim, K. Alghathbar, "Cryptanalysis and security enhancement of a more efficient and secure dynamic ID-based remote user authentication scheme," *Computer Communications*, vol. 34, no. 3, pp. 305–309, 2011.
- [8] S. K. Kim, M. G. Chung, "More secure remote user authentication scheme," *Computer Communications*, vol. 32, no. 6, pp. 1018–1021, 2009.
- [9] T. H. Kim, C. Kim, I. Park, "Side channel analysis attacks using AM demodulation on commercial smart cards with SEED," *Journal of Systems and Software*, vol. 85, no. 12, pp. 2899–2908, 2012.
- [10] Y. Lee, H. Kim, "Anonymous password-based authenticated key agreement scheme with non-temper resistant smart cards," *International Journal of Security and Its Applications*, vol. 9, no. 11, pp. 419–428, 2015.
- [11] C. T. Li, C. C. Lee, C. J. Liu, C. W. Lee, "A robust remote user authentication scheme against smart card security breach," in *Proceedings of 25th Annual IFIP Conference on Data and Applications Security and Privacy (DBSec'11)*, pp. 231–238, Richmond, VA, USA, 2011.
- [12] I. E. Liao, C. C. Lee, M. S. Hwang, "A password authentication scheme over insecure networks," *Journal of Computer and System Sciences*, vol. 72, no. 4, pp. 727–740, 2006.
- [13] S. Mangard, E. Oswald, T. Popp, *Power Analysis Attacks: Revealing the Secrets of Smart Cards*, Guaz University of technology Graz, Austria, 2007.
- [14] T. S. Messerges, E. A. Dabbish, R. H. Sloan, "Examining smart-card security under the threat of power analysis attacks," *IEEE Transactions on Computers*, vol. 51, no. 5, pp. 541–552, 2002.

Table 7: Security comparison

	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	C11	C12
[17]	✓	✓	✓		✓	✓	✓	✓	✓	✓	✓	✓
[10]		✓	✓	✓			✓	✓		✓		✓
[5]	✓	✓		✓			✓			✓		
[20]	✓	✓	✓				✓	✓		✓	✓	✓
[21]	✓						✓			✓		✓
Ours	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓

C1 No verifier table.

C2 Password can be chosen freely.

C3 The password cannot be derived by the privileged administrator of the server.

C4 The security of the scheme is not based on the tamper resistance assumption of the smart card.

C5 Resistance to known attacks, such as offline password guessing attack, replay attack, parallel session attack, denial of service attack, stolen verifier attack, user/server impersonation attack.

C6 The password cannot be broken by guessing attack even if the smart card is lost/stolen and compromised.

C7 Establish a common session key.

C8 The scheme is not prone to the problems of clock synchronization and time-delay

C9 the user can change the password locally without any interaction with the authentication server.

C10 Mutual authentication

C11 User anonymity.

C12 Forward secrecy.

- [15] E. O. Osei, J. B. Hayfron-Acquah, "Cloud computing login authentication redesign," *International Journal of Electronics and Information Engineering*, vol. 1, no. 1, pp. 1–8, 2014.
- [16] S. K. Sood, "Secure dynamic identity-based authentication scheme using smart cards," *Information Security Journal: A Global Perspective*, vol. 20, no. 2, pp. 67–77, 2011.
- [17] D. Wang, C. G. Ma, P. Wu, "Secure password-based remote user authentication scheme with non-tamper resistant smart cards," in *26th Annual IFIP Conference on Data and Applications Security and Privacy (DBSEC'12)*, pp. 114–121, Paris, France, July 11-13, 2012.
- [18] Y. Wang, J. Liu, F. Xiao, J. Dan, "A more efficient and secure dynamic ID-based remote user authentication scheme," *Computer Communications*, vol. 32, no. 4, pp. 583–585, 2009.
- [19] F. Wu, L. Xu, S. Kumari, X. Li and A. Alelaiwi, "A new authenticated key agreement scheme based on smart cards providing user anonymity with formal proof," *Security and Communication Networks*, vol. 8, no. 18, pp. 3847–3863, 2015.
- [20] Q. Xie, "Dynamic ID-based password authentication protocol with strong security against smart card lost attacks," in *First International Conference on Wireless Communications and Applications (ICWCA'11)*, pp. 412–418, Sanya, China, Aug. 1-3, 2011.
- [21] J. Xu, W. Zhu, D. Feng, "An improved smart card based password authentication scheme with provable security," *Computer Standards and Interfaces*, vol. 31, no. 4, pp. 723–728, 2009.

Biography

Min Wu biography. is working as an MS candidate in Applied Mathematics at Wuhan University, China. Her research interests information security and cryptographic protocol.

Jianhua Chen biography. received his BSc in Applied Mathematics from Harbin Institute of Technology, Harbin, China in 1983 and received MSc and PhD degree in Applied Mathematics from Wuhan University, Wuhan, China in 1989 and 1994, respectively. Currently, he is a Professor of Wuhan University. His current research interests include number theory, information security, and network security.

Ruibing Wang biography. is working as an MS candidate in Applied Mathematics at Wuhan University, China. Her research interests information security and cryptographic protocol.

A General Formal Framework of Analyzing Selective Disclosure Attribute-Based Credential Systems

Caimei Wang^{1,2}, Yan Xiong¹, Wenjuan Cheng³, Wenchao Huang¹, Huihua Xia¹,
Jianmeng Huang¹

(Corresponding author: Wenjuan Cheng)

School of Computer Science, University of Science and Technology of China¹
Elec-3 (Diansan) Building, West Campus of USTC, Huang Shan Road, Hefei, Anhui Province, China
Department of Computer Science, Hefei University²
Building 38, No.99, Jinxiu Avenue, Hefei, Anhui Province, China
School of Computer and Information, HeFei University of Technology³
193, Tunxi Road, Hefei, Anhui
(Email: cheng@ah.edu.cn)

(Received Nov. 16, 2016; revised and accepted Feb. 21 & Mar. 11, 2017)

Abstract

A selective disclosure attribute-based credential system (SDABCS) can provide a communication mechanism to protect both security and privacy in electronic communication, by issuing a kind of credential with attributes, which the user can disclose parts of attributes. We present a general framework for formally verification of SDABCS with applied Pi calculus, and provide three definitions of relevant security properties. The framework can implement secure communication among the user, service provider and trusted authority. Two important functions are implemented: the first allows the user to receive a credential encoded a list of attributes from a trusted authority; the second allows the user to convince a service provider with the credential. Particularly, the user can selectively reveal parts of the attributes according to the needs of service provider, while not revealing the rest of the attributes. In our experiments, we apply the framework to a concrete security protocol and successfully prove three security properties in the protocol using ProVerif.

Keywords: Attribute-based Credential; Formalize; General Framework; Selective Disclosure

1 Introduction

A selective disclosure attribute-based credential system [11, 17, 22] (SDABCS) allows a service provider to identify a user with a credential, which contains a list of attributes, from an authority. One of the most important characteristics in this system is that the user can selectively disclose

parts of the attributes in the credential to different service providers according to practical requirements. For the other undisclosed attributes, the user can generate cryptographic commitments to encode attributes which can be verified without revealing the attribute information. Authentication is one of the potential applications of this system. For example, *Alice* gets a credential C_{cre} with several attributes from an authority and generates two different presentation proofs, P_1 and P_2 , according to different attributes. Then she can access services provided by provider $S1$ under P_1 , and by $S2$ under P_2 , while no one can find that the two presentation proofs belong to the same user.

Although a lot of scholars study SDABCS [8, 14], so far there is no an effective formal verification framework to verify the correctness of the system and the related security properties. Actually, organizations and individuals are now paying more attention to the design of secure system. The formal analysis is a state-of-the-art method used to analyze security protocols and systems. There are many researches on the formal analysis and verification of security protocols [2, 4, 19, 23]. Li et al. propose a general symbolic model for anonymous credential protocols and make formal definitions of a few critical security properties [13]. Shao et al. conduct a formal analysis and verification of the enhanced authorization mechanism under TPM 2.0 API [20]. In [3], zero knowledge proof is applied to a simplified Direct Anonymous Attestation protocol, which enables remote authentication of *TPM* while preserving the user's privacy at the same time. They present security proof and report a novel attack. However, existing formal work considers neither the feature of selectively

disclosed attributes nor the credential protected by hardware.

In order to carry out an analysis of SDABCS, we propose a general framework of formally verifying the SD-ABCS. In our framework, a user can authenticate herself to a service provider by disclosing parts of her attributes, while preserving the privacy of the undisclosed ones. All attributes are encoded in the credential issued by an authority. In order to increase the security, we consider the situation that the credential is protected by the security hardware. The complexity of the system makes the formalized process involve many parameters. As a result, it is challenging to reasonably construct all the functions and equations with applied Pi calculus [9].

As a case study, we formally verify an innovative cryptographic protocol named U-Prove protocol [15, 18], whose SDK has been released by Microsoft and then integrated into a range of its own identity products. We first give the detailed description of the formal analysis, then prove the authenticity properties by setting correspondence assertions and prove the untraceability of U-Prove token by observational equivalence using ProVerif.

To the best of our knowledge, this is the first formal and automated verification of the SDABCS.

Contributions. The contributions of this paper are threefold: first, we put forward a general model framework of a SDABCS in applied Pi calculus, and provide three formal definition of security properties. It is meaningful to reduce the workload of verification of attribute-based credential system. Second, we apply our framework to the U-Prove protocol and give the formalized description of it. Finally, we prove that the protocol satisfies the relevant security properties by Proverif.

Outline of the Paper. In Section 2, applied Pi calculus is reviewed. In Section 3, we propose our general model framework of the SDABCS and the definitions of relevant security property. In Section 4, after detailed analysis of the U-Prove protocol, we apply the framework to verify the protocol, and successfully prove the three security properties using Proverif. In Section 5, conclusion and future work are presented.

2 Review of Applied Pi Calculus

2.1 Syntax and Semantics

The applied Pi calculus [3, 12, 21] is a language for describing and analyzing security protocols. Now we briefly review the syntax and operational semantics of the applied Pi calculus, and define the additional notation used in our paper. The syntax of the applied Pi calculus is given in Table 1.

The syntax of the calculus is composed of terms and processes. Terms are defined by a signature Σ which consists of a finite set of function symbols with arbitrary arity. Every function symbol means a primitive used by security protocols, and all symbols are divided into two finite sets of constructor and destructor symbols. Both two symbols

Table 1: Syntax of the applied Pi calculus

$M, N, F, Z ::=$	Terms
s, k, \dots, a, b	names
x, y, z	vars
$f(M_1, \dots, M_k)$ $f \in \Sigma$ and k is the arity of f	function
$P, Q ::=$	Processes
0	null process
$P \mid Q$	parallel composition
$vn.P$	name restriction
$!P$	replication
$u(x).P$	message input
$\bar{u}\langle N \rangle.P$	message output
if $M = N$ then P else Q	conditional

are built from an infinite set of names, and an infinite set of variables, e.g., encryption function, decryption function, digital signature function, etc. Usually, constructors are used to generate terms which model primitives used by protocols, while destructors are used to handle terms generated by constructors.

A process is a set of programs, which are connected via channels. That means two processes can communicate with each other by such channels. The executing program is known as the active program. An active program can execute operation and send messages to another program. A process or an extended process with a hole is called context, marked as $C[_]$.

In Pi calculus, the grammar of processes is defined as follows: the null process 0 does nothing; $P \mid Q$ is the parallel composition of process P and Q , used to express participants of a protocol running in parallel; the process $vn.P$ is used to produce a fresh name n and then behaves as P , in which the restricted name n is binded inside P ; replication process is the infinite compositions $P \mid P \mid \dots$, which means there are infinite copies of P running in parallel; in $u(x).P$ process, a message x can be received from the channel u , then the process behaves as P ; on the contrary, in $\bar{u}\langle N \rangle.P$ process, a message N can be sent to the channel u , and then process behaves as P ; the conditional process if $M = N$ then P else Q means that when equation holds process behaves as P , otherwise Q , when Q is null, we always abbreviate it as if $M = N$ then P .

A protocol P consists of a set of agents and channels, agents can communicate over the channels. Every agent runs a set of programs, and a program is an honest program only if it follows the protocol.

We equip the terms with an equational theory E , that is a finite set of equations of the form $M = N$ where $M, N \in \Sigma$.

2.2 Security Properties

This section presents two security properties that will be used in our paper.

Correspondence property is usually used to prove the

authentication of the participants in many protocols. We proof it by setting the events in protocol at different stages, and then verify identity according to the events occur successively relationship. So we can annotate processes with a set of events $\{e_1, e_2, \dots, e_n\}$ in a running protocol.

Definition 1 (Correspondence property). A correspondence property is used to express relationship between events with the form:

$$e_i < e_j \quad \text{where } i < j, \quad i, j \in \{1, 2, \dots, n\}$$

The property means that if an event e_i has happened then event e_j must have happened previously. Authentication can be captured as a correspondence property.

Zero-knowledge proof regarded as an important and most basic technology will be used in our system. Anyone can use zero-knowledge proof to communicate with other and convince the latter the given statement is true, without conveying any more information in addition to the statement. A formal definition of non-interactive zero-knowledge has been devised in [3].

Observational equivalence is the property that two or more underlying entities are indistinguishable on the basis of their observable implications. We define Observational Equivalence as the following description.

Definition 2 (Observational Equivalence). If there are two terms M and N satisfy $C[M]$ and $C[N]$ are both valid terms with the same value in all contexts $C[\cdot]$, then it is not possible, within the system, to distinguish the two terms. We call the observation equivalence as the largest symmetric relation between M and N .

3 General Model Framework of SDABCS

3.1 Modelling the Roles in Framework

A general model on the verification of a SDABCS allows a user to receive a credential with a list of attributes from a trusted party. This credential can be used to convince a service provider. The core feature of the system enables a user to select disclosable attributes, and generates the verifiable presentation proof for the undisclosed attributes, then the Verifier finishes the verification work.

There are three types of agents in our model: Issuer, Prover, and Verifier. The role of Issuer is an authority, who can generate and issue the credentials with a list of attributes containing an unforgeable digital signature by applying its private key I_{sk} . The role of Prover is a person in possession of a credential encoded a list of attributes, which can be optionally protected by a security hardware. Prover can control which attributes are revealed, and which attributes are generated presentation proof according to the needs of different service providers. The role of Verifier is a person who verifies the Issuer's signature in the credential, and the presentation proof of attributes generated by the Prover.

We set a process for each role in our model, and assume that the system is executed in a public network, where attackers can listen to, delete, forge and send all messages in the network, following the so-called Dolev-Yao model [10].

Prover process. When a Verifier relying on authentication or other identity-related attributes communicates with a Prover, the Prover must first demand a credential with attributes signed by an Issuer which is trusted by the Verifier. Then the Prover provides necessary attributes, and generates a cryptographic presented proof for the undisclosed attributes. Hence a Prover process can be defined as in Equation (1).

$$\text{Prover}P \stackrel{\text{def}}{=} \text{in } (c, IP). (P^{Ini}(CI, C_{sk}, C_{pk}) \quad (1) \\ | P^{getsig}(CI, I_{pk}) | P^{getp}(CI, AU_p, AC_p))$$

c models the public channel which is used to transmit all kinds of messages. IP models an unique identifier for the Issuer parameters including Issuer's public key. CI models an unique identifier of the credential. The process $P^{Ini}(CI, C_{sk}, C_{pk})$ models the Prover's behavior of generating the private key C_{sk} and the corresponding public key C_{pk} for the credential CI . The process $P^{getsig}(CI, I_{pk})$ models the Prover's behavior of getting the signature from Issuer I_{PK} using the blind operation to the credential. The process $P^{getp}(CI, AU_p, AC_p)$ models the Prover's behavior of generating presentation proof for the credential CI . AU_p models the proof of undisclosed attributes, and AC_p models the proof of needing to submit information commitment attributes. So presentation proof contained several parts as in Equation (2).

$$P^{getp}(CI, AU_p, AC_p) \stackrel{\text{def}}{=} P^{undisp}(CI, \quad (2) \\ AU) . P^{commitp}(CI, AC) . P^{TPMp}$$

where $P^{undisp}(CI, AU)$ models the Prover's behavior of generating presentation proof for undisclosed attributes AU in credential CI , and $P^{commitp}(CI, AC)$ models the behavior of generating presentation proof. If the token is protected by TPM 2.0, then P^{TPMp} models the Prover's behavior of generating presentation proof for TPM 2.0.

Issuer process. Correspondingly, an Issuer process consists of initializing issuer parameter modelled by the process $I^{Ini}(I_{pk}, IP)$, and signing the credential with specific attributes modelled by the process $I^{sign}(IP, CI)$. The process $I^{Ini}(I_{pk}, IP)$ models the behavior of generating issuer parameters by Issuer I_{pk} , an application-specific unique identifier for the issuer parameters is denoted as IP . The process $I^{sign}(I_{pk}, IP, CI)$ models Issuer I_{pk} signing the credential CI under IP . An Issuer process can be defined as in Equation (3).

$$\text{Issuer}P \stackrel{\text{def}}{=} I^{Ini}(I_{pk}, IP) | I^{sign}(IP, CI) \quad (3)$$

Verifier process. Verification of a credential by a Verifier is made up of four parts. So the verifier needs to do: firstly, verify the signature of credential CI from

the Issuer I_{pk} , which is modelled by $V^{Isig}(CI, I_{pk})$. Secondly, verify the proof of undisclosed attribute, which is modelled by $V^{undisp}(AUp, C_{pk})$. Thirdly, verify the proof of the committed attributes, which is modelled by $V^{commitp}(ACp, C_{pk})$. Lastly, if token is protected by the TPM 2.0, verify the proof of the TPM 2.0, which is modelled by V^{TPMp} . An Verifier process can be defined as in Equation (4).

$$\begin{aligned} \text{verifier}P \stackrel{\text{def}}{=} & V^{Isig}(CI, I_{pk}). V^{undisp}(AUp, \\ & C_{pk}). V^{commitp}(ACp, C_{pk}). V^{TPMp} \end{aligned} \quad (4)$$

3.2 Modules

According to the different functions implemented by the roles, the system needs to complete two modules. One is denoted as issuance module, the other is denoted as presentation proof module. A Prover can retrieve a credential encoded any kinds of attributes from an Issuer in an issuance module, and generate a presented proof for undisclosed attributes in the credential to a Verifier in a presentation proof module.

We model the issuance module process as an unbounded number of Prover processes and trusted Issuer processes running in parallel. The presentation proof module process is modelled as an unbounded number of Prover processes and Verifier processes running in parallel.

Definition 3 (Issuance module). *An Issuance Process is made up of four sub-processes, including the initialization process run by the Prover, the initialization process run by the Issuer, the process of producing and issuing credential run by the Issuer, the process of getting and blinding credential run by the Prover. Due to the process of blind in this protocol the Issuer never sees his own digital signature on the credential. The composition of the processes in applied π shows as Equation (5).*

$$\begin{aligned} IP \stackrel{\text{def}}{=} & v\tilde{n}. (P^{Ini}(CI, C_{sk}, C_{pk}) \mid P^{getsig}(CI, \\ & I_{pk}) \mid I^{Ini}(I_{pk}, IP) \mid I^{sign}(IP, CI)) \end{aligned} \quad (5)$$

The restricted name \tilde{n} models the secrets shared between the Provers and the Issuer. The rest of the parameters have the same meaning as described elsewhere.

Definition 4 (Presentation Proof module). *A Presentation Proof Process is made up of two sub-processes, including the presentation proof generation run by the Prover and the presentation proof verification run by the Verifier. The composition of the processes in applied π shows as Equation (6).*

$$\begin{aligned} \text{show}P \stackrel{\text{def}}{=} & \text{in}(c, \text{sig}_t). P^{getp}(CI, AUp, ACp) \\ & \mid \text{in}(c, \langle AUp, ACp, TPMp \rangle > P^{\text{verifier}P} \end{aligned} \quad (6)$$

The restricted c models the public channel. The rest of the parameters means the same above.

3.3 Events and Properties

As mentioned in a lot of papers [1, 5], correspondence assertions can be used to prove many trace-based security properties among events. Here we summarize a set of events which can be used later.

- **PCreInf**(CI, Cre_{sk}): where CI is the value of the credential information field. It is used to encode Credential-specific information that is always disclosed to Verifier, such as a validity period, credential usage restrictions and so on. Cre_{sk} is the private key of the credential, which is generated in the issuance protocol and should be kept secret. This event is executed after the Prover to generate the private key of the credential with the value of CI in the P^{Ini} process.
- **IssueCre**(CI, I_{sk}): where CI is the same as above. I_{sk} is the private key of the Issuer. This event is executed before the Issuer sends credential with the value of CI signed by himself.
- **PgetCre**(Cre_{pk}, I_{pk}): where Cre_{pk} is the public key of a credential corresponding to its private key Cre_{sk} . I_{pk} is the public key of the Issuer corresponding to its private key I_{sk} . This event is executed after the Prover to generates presentation proof for the credential with Cre_{pk} .
- **VerifiedCre**($Cre_{pk}, proof, I_{pk}$): where Cre_{pk} , $proof$ and I_{pk} are the same as above. $proof$ is one of the parameters needs to be proved. This event is executed after successful validation by the Verifier.

According to the definition of Section 2.2, we provide three definitions of basic security properties which the general model framework needs to meet.

Definition 5 (Authenticity of the Issuance Protocol). *Given processes as follows.*

$$\langle P^{getsig}(CI, I_{pk}) \mid I^{sig}(CI, I_{sk}) \rangle$$

authenticity of the Issuance Protocol is satisfied if the process satisfies the following property:

$$\begin{aligned} \forall(CI, I_{sk}, I_{pk}), \exists\{P^{getCre}(CI, I_{pk}) \\ \implies \text{IssueCre}(CI, I_{sk})\} \end{aligned}$$

This property shows that when a Prover gets a token signed by an Issuer with public key I_{pk} , the Issuer has actually signed with the corresponding private key I_{sk} and issued that credential.

Definition 6 (Authenticity of the Presentation Protocol). *Given processes as follows.*

$$P^{getsig}(CI, I_{pk}) \mid P^{\text{verifier}P}$$

authenticity of the Presentation Protocol is satisfied if the process satisfies the following correspondence property.

$$\begin{aligned} & \forall (CI, Cre_pk, Cre_sk), \\ & \exists \{VerifiedCrek(Cre_pk, proof, I_pk) \implies \\ & PCreInf(CI, Cre_sk) PgetCre(Cre_pk, I_pk)\} \end{aligned}$$

This property shows that if there is a Verifier who can complete the validation of *proof* signed by the Issuer and *Cre_sk*, we can say not only an Issuer with public key *I_pk* has actually signed and issued that credential with the corresponding private key *I_sk*, but also the Prover has got that credential.

Definition 7 (Untraceability of Credential). *Assume the same attributes CI encoded into two different credentials (CI₁, CI₂) by the Prover, if the issuer cannot identify two credential from the same user, we call this feature as untraceability.*

This property shows that when a user provides two credentials issued by an issuer, the issuer cannot judge whether the two certificates are from the same user.

4 Case Study: U-Prove Protocol

4.1 Detail of the U-Prove Protocol

U-Prove is an innovative cryptographic technology, its core is a U-Prove token [18] with any type of application-specific attributes. There are three types of agents in the protocol: *Prover* (*P*), *Issuer* (*I*), *Verifier* (*V*). Each U-Prove token can be seen as a credential of a *Prover* (*P*). It can be optionally protected by a trusted hardware such as security chip TPM 2.0. Its prototype has been first introduced in [7] by Liquan Chen. The trusted hardware in this paper refers to TPM 2.0. In the rest of the paper, we use the following notation: Let $G_q = \langle g \rangle$ be a cyclic group of prime order q and g be a generator.

There are two sub-protocols in U-Prove protocol: issuance protocol and presentation proof protocol. The former mainly generates and issues a U-Prove token. The latter mainly produces and verifies a presentation proof about the undisclosed attributes in the U-Prove token.

The Issuance Protocol.

Assume *I* and *P* agree on the application-specific attributes (A_1, \dots, A_n), the value of the token information field *TI* is used to encode token-specific information. *P* wants to get a U-Prove token with attributes $A_i, i \in (1, \dots, n)$ from *I*, both parties must communicate with each other under the issuance protocol.

The protocol consists of the following steps:

- 1) *I* generates the issuer parameters denoted as *IP*.

$$IP = \{(g_0, g_1, \dots, g_n, g_t), (e_0, e_1, \dots, e_n)\}$$

where $(g_0, g_1, \dots, g_n, g_t)$ represents *I*'s public key and satisfies the equation $g_0 = g^{y_0}$, which y_0 is *I*'s private key. The rest of g_i values must be random generators of G_q .

- 2) *I* and *P* complete the precomputation respectively: $\gamma = (g_0 g_1^{x_1} \dots g_n^{x_n} g_t^{x_t} h_d)$, where (x_1, \dots, x_n) are the operation results, according to the values in the list of (e_0, e_1, \dots, e_n) . The value of e_i is 0 or 1. $x_i = H(A_i)$ when $e_i = 1$, otherwise $x_i = A_i$. H is a collision-resistant hash function. The parameter h_d means that the credential needs to be protected by TPM 2.0 with private key x_d and the corresponding public key $h_d = g_d^{x_d}$. The parameter g_d is one of the random generators of G_q .

- 3) Signature process of *I* includes two steps: commit computation and signature computation. The former contains parameters: $\sigma_a = g^w$, $\sigma_b = \gamma^w$, w is a random number. The latter contains parameter: $\sigma_z = \gamma^{y_0}$ and sends $(\sigma_a, \sigma_b, \sigma_z)$ to *P*.

- 4) The main job of *P* consists of two parts: one is generating a pair of keys for the U-Prove token, the other is masking the parameters come from *I* which prevents *I* from seeing the value of its signature. *P* generates a random α and computes $h: h = (g_0 g_1^{x_1} \dots g_n^{x_n} g_t^{x_t} h_d)^\alpha$. The value of α^{-1} is regarded as the U-Prove token's private key and the public key is h . After receiving the message $(\sigma_a, \sigma_b, \sigma_z)$, *P* produces two blind factors β_1 and β_2 to mask the three parameters respectively, then gets $(\sigma'_a, \sigma'_b, \sigma'_z)$ and σ'_c , using the following formula respectively:

$$\begin{aligned} \sigma'_a &= g_0^{\beta_1} g^{\beta_2} \sigma_a, \\ \sigma'_z &= \sigma_z^\alpha, \\ \sigma'_b &= \sigma_z^{\beta_1} h^{\beta_2} \sigma_b^\alpha \\ \sigma'_c &= H(h, PI, \sigma'_z, \sigma'_a, \sigma'_b) \rightarrow Z_q, \\ \sigma_c &= \sigma_c + \beta_1 \text{ mod } q \end{aligned}$$

where *PI* is the Prover information field produced by *P*, which is always revealed during presentation protocol. Then *P* sends σ_c to *I*.

- 5) After *I* got the parameter σ_c , *I* generates and conveys the signature σ_c with private key y_0 to *P* by the Schnorr signature scheme: $\sigma_r = \sigma_c y_0 + w \text{ mod } q$.
- 6) After *P* received the parameter σ_r , *P* masks σ_r with β_2 as follow: $\sigma'_r = \sigma_r + \beta_2 \text{ mod } q$. Then *P* verifies the validity of the signature, as in Equation (7).

$$\sigma'_a \sigma'_b = (gh)^{\sigma'_r} (g_0 \sigma'_z)^{-\sigma'_c} \quad (7)$$

P gets a valid U-Prove token denoted as follows if result is valid.

$$\mathcal{T} = h, TI, PI, \sigma'_z, \sigma'_c, \sigma'_r$$

The Presentation Proof Protocol.

After getting an issued U-Prove token (\mathcal{T}) with a pair of keys (α^{-1}, h) , where α^{-1} is secret as private key, h is disclosed to *V* without revealing to *I* in the issuance

Protocol as public key. When using \mathcal{T} , P can selectively disclose parts of attributes to V , and create a presentation proof for undisclosed attributes. \mathcal{T} can be protected by the TPM 2.0 or not. The usage of \mathcal{T} based on the TPM 2.0 protection will be discussed in this section.

Generate Presentation.

P has the right to determine the following parameters: the indices of disclosed attributes is denoted as $D \subset \{1, \dots, n\}$, the indices of undisclosed attributes is denoted as $U \subset \{1, \dots, n\} - D$, the indices of committed attributes is denoted as $C \subset U$. P can not only generate a pseudonym for \mathcal{T} , but also specify the pseudonym derived from a specific scope s .

P can generate the presentation proof with the TPM 2.0 as follows:

- 1) P sends s to the TPM 2.0.
- 2) After receiving s , TPM 2.0 generates w'_d at random, then computes: $a_d = g_d^{w'_d}$, $g_s = GenEle(s)$, $a'_p = g_s^{w'_d}$, $P'_s = g_s^{x_d}$ and sends (a_d, a'_p, P_s) to P .
- 3) P computes each x_i , and generates w_0 , w_d , for $i \in U$. P generates w_i at random, for $i \in C$. P generates \tilde{o}_i, \tilde{w}_i at random. Then computes equations as follow.

$$a = H(h^{h_0}(\prod_{i \in U} g_i^{w_i}) g_d^{w_d} a_d),$$

$$g_s = GenEle(s), \quad a_p = H(g_s^{w_p} a'_p),$$

$$\tilde{c}_i = g^{x_i} g_1^{\tilde{o}_i}, \quad \tilde{a}_i = H(g^{w_i} g_1^{\tilde{w}_i}),$$

$$c_p = H(a, D, x_{i, i \in D}, C, \{\tilde{c}\}_{i, i \in C}, \{\tilde{a}\}_{i, i \in C}, a_p, P_s, m),$$

$$c = H(c_p, m_d), r_0 = c\alpha^{-1} + w_0,$$

$$r_i = -cx_i + w_{i, i \in U}$$
 sends (c_p, m_d) to the TPM 2.0.
- 4) TPM 2.0 computes and sends the response r'_d :

$$c = H(c_p, m_d), r'_d = -cx_d + w'_d.$$

- 5) P completes the following operation and eventually gets the presentation proof about \mathcal{T} :

$$PP = \langle A_{i, i \in D}, a, (a_p, P_s), r_0, r_{i, i \in U}, r_d, \{\tilde{c}_i, \tilde{a}_i, \tilde{r}_i\}_{i \in C} \rangle.$$

where: $r_d = r'_d + w_d$, $\tilde{r}_i = -c\tilde{o}_i + \tilde{w}_i$ ($i \in C$)

Verify Presentation. Given a presented \mathcal{T} , V can check it without any secret information. V first gets the input parameters:

$$\begin{aligned} x_i &= cxi(IP, A_i) \text{ for each } i \in D \\ c_p &= H(a, D, \{x_{i, i \in D}, C, \{\tilde{c}_i, i \in C, \{\tilde{a}_i, i \in C, a_p, P_s, m\} \\ c &= H(c_p, m_d). \end{aligned}$$

Then V computes the following equations:

$$\begin{aligned} a &= H((g_0 g_t^{x_t} \prod_{i \in D} g_i^{x_i})^{-c} h^{r_0} (\prod_{i \in U} g_i^{r_i}) g_d^{r_d}) \\ g_s &= GenEle(s), \quad a_p = H(P_s^c g_s^{r_d}), \\ \tilde{a}_i &= H(\tilde{c}_i^c g_1^{r_i} g_1^{\tilde{r}_i}) (i \in C). \end{aligned}$$

If the above equations are proved, the authentication is successful.

4.2 Modelling the U-Prove Protocol

According to the previous framework, we model the U-Prove Protocol with the applied Pi calculus. This calculus is an extension of the Pi calculus with function symbols which satisfy particular equations. We design function symbols and an equational theory for modelling the U-Prove protocol. All the parameters of this section have the same meaning as Subsection 4.2.

According to our model in Section 3, we define three processes according to different roles:

- *Prover* process (written as \mathcal{P}) with three sub-processes $Pini$, $Pgetsig$, $Pgetp$.
- *Issuer* process (written as \mathcal{I}) with two sub-processes $Iini$ and $Isig$.
- *Verifier* process (written as \mathcal{V}) with one sub-process *verifier*, TPM 2.0 process (written as $\tilde{\mathcal{T}}$) with one sub-process TPM 2.0.

We define a public channel c to represent a public network through which all messages are transmitted. How I and P agree on the contents of the issued U-Prove token, and how TPM 2.0 provides its public key are outside the scope of this paper.

Functions. We define the relevant functions in this paper with respect to the signature

$$\begin{aligned} \Sigma = \{ &c/2, \text{sign}/2, \text{Schsign}/2, \text{getgt}/1, \text{getgU}/1, \\ &\text{getgD}/1, \text{getgc}/1, \text{getAD}/1, \&\text{getAU}/1, \\ &\text{getAC}/1, \text{cxt}/3, \text{cxi}/2, b/2, bc/2, b1/2, b2/2 \} \end{aligned}$$

, where,

- $\{c/2\}$ models generating a commitment value.
- $\{\text{sign}/2\}$ models signing messages.
- $\{\text{Schsign}/2\}$ models getting a Schnorr signature.
- $\{\text{getgt}/1, \text{getgU}/1, \text{getgD}/1, \text{getgc}/1\}$ models derivation functions to derive Issuer's public key.
- $\{\text{getAD}/1, \text{getAU}/1, \text{getAC}/1\}$ models derivation functions to derive all attributes by Prover and Issuer.
- $\{\text{cxt}/3, \text{cxi}/2\}$ models getting the value of parameters used to compute the γ .
- $\{b/2, bc/2, b1/2, b2/2\}$ models the blind processes of commitments in different situations.

Equations. After setting all functions, we specify an equational theory in terms of a convergent rewriting system. This theory is suitable for ProVerif.

- 1) $b2(\text{sign}(\gamma, y0), \alpha) = \text{blindsign}(\gamma, \alpha, y0)$
- 2) $bc(c(\gamma, w), \alpha) = bc2(\gamma, \alpha, w)$
- 3) $bc(c(\gamma, \alpha), \beta_1, \beta_2) = bc3(\gamma, \alpha, \beta_1, \beta_2)$

- 4) $bc(bc(\gamma, \alpha, w), \beta_1, \beta_2) = bc4(\gamma, \alpha, w, \beta_1, \beta_2)$
- 5) $verif(bc3(\gamma, \alpha, \beta_1, \beta_2), b2(schsign(b(H(h, bc3(\gamma, \alpha, \beta_1, \beta_2), \sigma_{b1}, \sigma_{z1}), \beta_1), y0), \beta_2), pk(g, y0)) = true$

The four equations are convergence equations and form a convergent rewriting system when oriented from left to right. The last equation can be used to check parameters.

Modelling of the Issuance Protocol.

We denote IP as the unique identifier of I parameters under which a \mathcal{T} is issued, all *Issuer* parameters are derived from IP using the derivation functions such as $getgt(IP)$, $getgU(IP)$, $getgD(IP)$, $getgC(IP)$. We denote TI as the unique identifier of the \mathcal{T} , all attributes are derived from TI , including disclosed attributes, undisclosed attributes and committed attributes, which are modelled by $getAD(TI)$, $getAU(TI)$, $getAC(TI)$ respectively.

The signature is generated by I under the Schnorr type of signature scheme [7]. The generation of the signature by I is divided into two parts: generation of commitment using a random number, and generation of signature using the private key. The function symbol $c/2$ represents the commitment constructor, and $sign/2$ represents the signature constructor.

\mathcal{I} interacts with \mathcal{P} as the following.

- To initiate a new session, \mathcal{I} sends the messages $\bar{c}(sign(\gamma, y_0), c(g, w), c(\gamma, w))$ with a public parameter $sign(\gamma, y_0)$ and two commitments $c(g, w), c(\gamma, w)$.
- After receiving $\{sign(\gamma, y_0), c(g, w), c(\gamma, w)\}$, \mathcal{P} calculates and sends out parameter σ_c .
- After receiving $\{\sigma_c\}$, \mathcal{I} begins to generate the Schnorr signature by the operation of $\sigma_z = schsign(\sigma_c, y_0)$, then sends out $\{\sigma_z\}$.
- After receiving $\{\sigma_z\}$, \mathcal{P} begins to verify and blind the signature, and finally get the blind signature token by \mathcal{I} .

Modelling of the Presentation Protocol.

In this paragraph, we assume that the U-Prove token is protected by TPM 2.0 and V needs commitments about some undisclosed attributes. P can only complete the presentation proof generation under the help of TPM 2.0.

P needs to provide the disclosed attributes, I 's signature, the token's public key and all presentation proofs. Presentation proofs include presentation proof for undisclosed attributes, presentation proof for committed attributes and presentation proof for information protected by TPM 2.0, denoted as a, \tilde{a}_i, a_p respectively. We use $getAD(TI)$ to get the disclosed attributes (denoted as AD) and $b(\gamma, \alpha)$ to get the token's public key (denoted as h) in our codes.

\mathcal{P} interacts with $\tilde{\mathcal{T}}$ as the following.

- $\tilde{\mathcal{T}}$ generates parameters and sends out $\{c(gd, wd1), c(gs, wd1), getgs(descG)\}$.

- \mathcal{P} generates the related presentation proofs and sends out $\{h, a, ap, ai, r0, rU, ri, rd\}$.

where h models \mathcal{T} 's public key, a models presentation proof of the undisclosed attributes, ap models presentation proof of the protection of the token by TPM 2.0, ai models presentation proof of the commitments which is needed by V , $r0$ models blind signature signed with token's private key, rU models blind signature of each undisclosed attributes, ri models blind signature of each committed attributes, rd models blind signature signed with TPM's private key.

4.3 Security Analysis of the U-Prove Protocol

4.3.1 Authenticity of the Issuance Protocol

The first property we would like to model is the authenticity of the issuance protocol: if P reaches the end of the protocol and she believes she has got a U-Prove token issued by I , then I has really issued the token. To prove this property we annotate processes with some events, marking important stages reached by the protocol which do not have effect on the execution of the processes.

There are a lot of test events in our experiments, here we list three events, which are denoted as follows.

- 1) **Issuetoken** (TI, g, y_0). I executes this event before signing the token with the private key y_0 and issuing a token with *identifier* TI .
- 2) **Gettoken** (TI, σ_{r1}, g_0). P executes this event after receiving the token signed by I with the public key g_0 .
- 3) **Bisign** (σ_{r1}). P executes this event after blinding the token issued by I .

Given processes as follows.

$$\langle I^{Ini}(IP, y_0) \mid P^{Ini}(TI, IP, g_0) \mid I^{sig}(TI, IP, y_0) \mid P^{getproof}(h, \alpha) \rangle$$

According to the definition 5. We have verified the authenticity of the issuance protocol by the following two trace properties with ProVerif.

- 1) $\forall(TI, g, y_0), \exists\{Gettoken(TI, \sigma_{r1}, pk(g, y_0)) \implies Issuetoken(TI, g, y_0)\}$,
- 2) $\forall(TI, \sigma_{r1}, g, y_0), \exists\{Gettoken(TI, \sigma_{r1}, pk(g, y_0)) \implies Issuetoken(TI, g, y_0) \vee Bisign(\sigma_{r1})\}$

The first result shows that when P gets a token signed by I with public key $pk(g, y_0)$, then I has actually signed with the corresponding private key y_0 and issued that token. The second results shows that when P gets a token signed by I with public key $pk(g, y_0)$, not only the I has actually signed with the corresponding private key y_0 and issued that token, but also P has blinded the token.

In our experiment, we set up the event *Issuetoken* in the process $I^{sig}(TI, IP, y_0)$, event *Gettoken* in the process $P^{Ini}(TI, IP, g_0)$, event *Bisign* in the process $P^{getproof}(h, \alpha)$ respectively. The following results are obtained after running the program.

- 1) RESULT $event(Gettoken(h_{.162}, pk(g_{.163}, y0_{.164}))) \implies event(Issuetoken(TI_{.161}, g_{.163}, y0_{.164}))$ is true.
- 2) RESULT $event(Gettoken(h_{.162}, pk(g_{.163}, y0_{.164}))) \implies event(Bisign(TI_{.161}, h_{.162})) \ \&\& \ event(Issuetoken(TI_{.161}, g_{.163}, y0_{.164}))$ is true.

4.3.2 Authenticity of the Presentation Protocol

We model the authenticity of the presentation protocol: if V reaches the end of the protocol with the presented U-Prove token, then P has really sent out the presented token signed by I .

As above, we annotate processes with three events and mark important stages reached by the protocol. Three events are denoted as follows.

- 1) **Gettokenpk** (h, α, gU, xU). P executes this event after the initialization of parameters, and gets the U-Prove token public key as token's identification.
- 2) **Evpproof** (h, a). P executes this event before she sends out the generation proof of the token.
- 3) **Verifok** ($h, a, multpk2(pk(h, inverse(\alpha)))$). V executes this event after she verifies the validity of the token's presentation proof.

Given processes as follows.

$$P^{getsig}(g_0, g, gU, gd) \mid P^{getproof}(h, \alpha) \mid verifier(g, g_0)$$

According to the definition 6. We have verified the authenticity of the Presentation Protocol with ProVerif by the following trace property.

$$\begin{aligned} & \forall(h, a, \alpha, gU, xU) : \\ & \exists\{Verifok(h, a, multpk2(pk(h, inverse(\alpha)), \\ & \quad pk(gU, xU))) \implies Evpproof(h, a) \\ & \quad \vee Gettokenpk(h, \alpha, gU, xU)\}. \end{aligned}$$

In our experiment, We set up the event *Gettokenpk* in the process $P^{getsig}(g_0, g, gU, gd)$, event *Evpproof* in the process $P^{getproof}(h, \alpha)$, event *VerifOK* in the process $verifier(g, g_0)$ respectively. The result shows that once validation is completed by V , then not only a valid I has actually signed and issued that token, but also P has blinded the token issued by I .

The following result is obtained after running the program.

$$\begin{aligned} & \text{RESULT } event(VerifOK(h_{.213}, a_{.214}, multpk2(pk(h_{.213}, \\ & \quad inverse(alpha_{.215})), pk(gU_{.216}, xU_{.217})))) \\ & \implies (event(Evpproof(h_{.213}, a_{.214})) \&\& event(Gettokenpk \\ & \quad (h_{.213}, alpha_{.215}, gU_{.216}, xU_{.217}))) \text{ is true.} \end{aligned}$$

4.3.3 Untraceability of U-Prove Token

In order to protect the identity information about individuals. During the Issuance Protocol, the Issuer uses blind signature rather than conventional RSA or DSA signature, and issuance is a three-leg interactive protocol enabling the Prover to hide certain token elements from the Issuer. This makes the Issuer never see its own digital signature on an issued U-Prove token, and never see the public key of the U-Prove token.

The blind signature scheme provides a strong privacy guarantee for the Issuance Protocol by untraceability property: the Issuer and all Verifiers cannot learn even a single bit of information beyond what can be inferred from the disclosed attributes in presented U-Prove tokens, even if they would collude from the outset.

According to Definition 7. We have verified the untraceability of U-Prove token by the observational equivalence between two processes $P1, P2$, where in Pi the Prover gets and provides the token \mathcal{T}_i . In our design we define a natural formulation $P1$ and $P2$ as follows:

$$\begin{aligned} P_i := & \\ & let(TI, gd) = (TI1, gd1) \text{ in } PIni(TI, gd) \mid \\ & let(TI, gd) = (TI2, gd1) \text{ in } PIni(TI, gd) \mid \\ & in(sc2, (hi, \alpha i)); \\ & let(TI, h, \alpha) = (TI1, hi, \alpha i) \text{ in } Pgetsig(TI, h, \alpha). \end{aligned}$$

We get the following result in our experimental.

RESULT Observational equivalence is true (bad not derivable).

5 Conclusion and Future Work

In this paper, we provide a general model framework on the SDABCS, and as a case study, provide the first formal analysis of U-Prove protocol. We give the detailed definitions about authenticity in each sub-protocols and untraceability of U-Prove token. Authenticities are expressed as a correspondence property and untraceability is proved by observational equivalence. All of the security properties are suitable for ProVerif [6].

As an innovative cryptographic technology, in addition to the properties proven in our paper, there are a lot of contents worth studying in SDABCS, such as revocable [16], reusable and so on. In particular, the properties of accountability has attract the attention of experts in recent years, we think studying the accountability in the U-Prove protocol is necessary.

Acknowledgments

The research is supported by National Natural Science Foundation of China under Grant No.61572453, No.6120-2404, No.61520106007, No.61170233, No.61232018, No.61-572454, Natural Science in Colleges and Universities in

Anhui Province under Grant No.KJ2015A257, and Anhui Provincial Natural Science Foundation under Grant No.1508085SQF215. We gratefully acknowledge the anonymous reviewers for our valuable comments.

References

- [1] M. Abadi, B. Blanchet, and C. Fournet, “Just fast keying in the pi calculus,” *ACM Transactions on Information and System Security*, vol. 10, no. 3, pp. 9, 2007.
- [2] R. Amin, “Cryptanalysis and efficient dynamic id based remote user authentication scheme in multi-server environment using smart card,” *International Journal of Network Security*, vol. 18, no. 1, pp. 172–181, 2016.
- [3] M. Backes, M. Maffei, and D. Unruh, “Zero-knowledge in the applied pi-calculus and automated verification of the direct anonymous attestation protocol,” in *IEEE Symposium on Security and Privacy (SP’08)*, pp. 202–215, 2008.
- [4] G. Barthe, B. Grégoire, and S. Zanella B., “Formal certification of code-based cryptographic proofs,” *ACM SIGPLAN Notices*, vol. 44, no. 1, pp. 90–101, 2009.
- [5] B. Blanchet and A. Chaudhuri, “Automated formal analysis of a protocol for secure file sharing on untrusted storage,” in *IEEE Symposium on Security and Privacy (SP’08)*, pp. 417–431, 2008.
- [6] B. Blanchet, B. Smyth, and V. Cheval, *Proverif 1.90: Automatic Cryptographic Protocol Verifier, User Manual and Tutorial*, 2014.
- [7] L. Chen and J. Li, “Flexible and scalable digital signatures in tpm 2.0,” in *Proceedings of the 2013 ACM SIGSAC conference on Computer & Communications Security*, pp. 37–48, 2013.
- [8] L. Chen and R. Urian, “Daa-a: Direct anonymous attestation with attributes,” in *International Conference on Trust and Trustworthy Computing*, pp. 228–245, 2015.
- [9] S. Delaune, M. Ryan, and B. Smyth, “Automatic verification of privacy properties in the applied pi calculus,” in *IFIP International Conference on Trust Management*, pp. 263–278, 2008.
- [10] D. Dolev and A. Yao, “On the security of public key protocols,” *IEEE Transactions on Information Theory*, vol. 29, no. 2, pp. 198–208, 1983.
- [11] R. Gay, I. Keremidis, and H. Wee, “Communication complexity of conditional disclosure of secrets and attribute-based encryption,” in *Annual Cryptology Conference*, pp. 485–502, 2015.
- [12] J. Goubault-Larrecq, C. Palamidessi, and A. Troina, “A probabilistic applied pi-calculus,” in *Asian Symposium on Programming Languages and Systems*, pp. 175–190, 2007.
- [13] X. Li, Y. Zhang, and Y. Deng, “Verifying anonymous credential systems in applied pi calculus,” in *Cryptology and Network Security*, pp. 209–225, 2009.
- [14] W. Lueks, G. Alpár, J. Hoepman, and P. Vullers, “Fast revocation of attribute-based credentials for both users and verifiers,” in *IFIP International Information Security Conference*, pp. 463–478, 2015.
- [15] W. Mostowski and P. Vullers, “Efficient u-prove implementation for anonymous credentials on smart cards,” in *Security and Privacy in Communication Networks*, pp. 243–260, 2012.
- [16] L. Nguyen and C. Paquin, “U-prove designated-verifier accumulator revocation extension,” Technical Report MSR-TR-2014-85, Microsoft Research, 2014.
- [17] T. Okamoto and K. Takashima, “Efficient attribute-based signatures for non-monotone predicates in the standard model,” *IEEE Transactions on Cloud Computing*, vol. 2, no. 4, pp. 409–421, 2014.
- [18] C. Paquin, *U-prove Technology Overview v1*, 2013.
- [19] H. Patel, D. Jinwala, M. Highway, M. Bhandu, and S. Ichchhanath, “Automated analysis of internet key exchange protocol v2 for denial of service attacks,” *International Journal of Network Security*, vol. 17, no. 1, pp. 66–71, 2015.
- [20] J. Shao, Y. Qin, D. Feng, and W. Wang, “Formal analysis of enhanced authorization in the tpm 2.0,” in *Proceedings of the 10th ACM Symposium on Information, Computer and Communications Security*, pp. 273–284, 2015.
- [21] F. Tiezzi and N. Yoshida, “Reversible session-based pi-calculus,” *Journal of Logical and Algebraic Methods in Programming*, vol. 84, no. 5, pp. 684–707, 2015.
- [22] P. Vullers and G. Alpár. “Efficient selective disclosure on smart cards using idemix,” in *Policies and Research in Identity Management*, pp. 53–67, 2013.
- [23] S. Yu, C. Wang, K. Ren, and W. Lou, “Attribute based data sharing with attribute revocation,” in *Proceedings of the 5th ACM Symposium on Information, Computer and Communications Security*, pp. 261–270, 2010.

Biography

Caimei Wang was born in 1978. She is a lecturer in Department of Computer Science and Technology, HeFei University. She also is a Ph.D. candidate of University of Science and Technology of China. Her main research interests include computer network, information security, and mobile computation. (Email: wangcmo@mail.ustc.edu.cn)

Yan Xiong was born in 1960. He is a professor in School of Computer Science and Technology, University of Science and Technology of China. His main research interests include distributed processing, mobile computation, computer network and information security. (Email: yxiong@ustc.edu.cn)

Wenjuan Cheng (corresponding author) was born in 1970. She is a professor in School of Computer and

Information, Hefei University of Technology. Her main research interests include computer network and information security, computer application technology. (Email: cheng@ah.edu.cn)

Wenchao Huang was born in 1982. He received both the B.S. and Ph.D. degrees in computer science from University of Science and Technology of China. He is associate professor in School of Computer Science and Technology, University of Science and Technology of China. His current research interests include mobile computing, information security, trusted computing and formal methods. (Email: huangwc@ustc.edu.cn)

Huihua Xia was born in 1994. He received the B.S. degree in computer science from University of Science and Technology of China. He is currently a PhD student in School of Computer Science and Technology, University of Science and Technology of China. His current research interests include data publishing and data privacy. (Email: download@mail.ustc.edu.cn)

Jianmeng Huang was born in 1991. He received the B.S. degree in computer science from University of Science and Technology of China in 2013. He is currently working towards the Ph.D. degree at the Department of Computer Science and Technology, University of Science and Technology of China. His current research interests include information security and mobile computing. (Email: mengh@mail.ustc.edu.cn).

A New Security Cloud Storage Data Encryption Scheme Based on Identity Proxy Re-encryption

Caihui Lan¹, Haifeng Li², Shoulin Yin³, and Lin Teng³
(Corresponding author: Haifeng Li)

College of Electronic and Information Engineering, Lanzhou City University¹
Lanzhou 730070, China

School of Software, Dalian University of Technology²

No.321, Tuqiang Street, Economy & Technology, Development Zone, Dalian, Liaoning 116620, P.R. China

(Email: lihaifeng8848@mail.dlut.edu.cn)

Software College, Shenyang Normal University³
Shenyang 110034, China

(Received Apr. 26, 2016; revised and accepted July 8 & Sept. 3, 2016)

Abstract

In the process of cloud data storage, data owner will encrypt data and upload it to the cloud, however, this method cannot support for encrypted data sharing. Especially, when data is shared with many users, the scalability is very weak. In order to solve this problem, we put forward a new security cloud storage data encryption scheme based on identity proxy re-encryption in this article. This scheme can flexibility share data with other users security without fully trusted cloud. For the detailed structure, we use a strong unforgeable signature scheme to make the transmuted ciphertext have publicly verification combined identity-based encryption. Furthermore, the transformed ciphertext has chosen-ciphertext security under the standard model. Because this new scheme can support fine-grained access control without using public key certificate and has better extensibility, so this scheme can be better applied into security cloud data sharing.

Keywords: Cloud Data Storage; Identity Proxy Re-encryption; Publicly Verification; Strong Unforgeable Signature

1 Introduction

Due to the rapid development of modern information technology, traditional data sharing way cannot satisfy the demand of social development. Cloud storage system [5,16] arises currently, and it can make users storage data in cloud at any time. Although cloud storage is very convenience for users, it may be insecurity stored in unbelievable third party. Therefore, it is necessary to ensure confidentiality, integrity and reliability of data.

In order to guarantee the confidentiality of data in the

cloud storage, users will encrypt data with encryption algorithm before uploading private information including advanced encryption standard [13], mixed encryption [7, 17], encryption based on attributes [9] and proxy re-encryption [1,11]. Han [4] proposed a privacy-preserving decentralized key-policy decentralized attribute-based encryption (ABE) scheme where each authority could issue secret keys to a user independently without knowing anything about his global identifier. Therefore, even if multiple authorities were corrupted, they could not collect the user's attributes by tracing his global identifier. Notably, the new scheme only required standard complexity assumptions and did not require any cooperation between the multiple authorities. Qiu [14] presented a new scheme which could avoid the collusion of proxy and delegatee and it improved the scheme of Chu and Tzeng while inheriting all useful properties such as unidirectionality and non-interactivity. In the new scheme, it got the security by using added secret parameter and changed the secret key and re-encryption key. Kgaikwad [8] created an efficient provable data possession method for distributed cloud storage, in which multiple cloud service providers were maintaining and storing client's data in cooperative way. This cooperatively working provable data possession method was based on indexing hierarchy & homomorphic variable response method.

In this paper, we propose an encryption scheme combining identity proxy re-encryption based on proxy re-encryption, which is fit for cloud data sharing. This scheme can flexibility share data with other users security without fully trusted cloud compared to general cloud storage access control schemes. We use a strong unforgeable signature scheme to make the transmuted ciphertext have publicly verification combined identity-based encryption. Furthermore, the transformed ciphertext has

chosen-ciphertext security under the standard model. Because this new scheme can support fine-grained access control without using public key certificate and has better extensibility and it also can filter malicious cipher, so this scheme can be better applied into security cloud data sharing. As we all know, there are two traditional ways to share data. One is that users encrypt data and put it into cloud. But cloud cannot effectively share data according to the requirement of users. Another one is that users directly put data into cloud and cloud server will handle the data with a fully credible cloud, which is impossible. Our scheme is flexible and convenient to realize data sharing, and it ensures control of sensitive data. What's more, new scheme avoids collusion attack at the same time. We ignore the system workload.

The rest of the paper is organized as follows: Section 2 introduces the transactional and cryptographic primitives that provide the foundation for the protocols presented in this work. Section 3 outlines the proposed schema to analyze detailed system model. The main contribution of the paper, i.e. the privacy preserving profiling protocols and security analysis are given in Section 4. Section 5 finally concludes the paper.

2 Preliminaries

2.1 Bilinear Map

A prime-order bilinear group generator is an algorithm GP that takes as input a security parameter λ and outputs a description $\Gamma = (p, G, G_T, e, g)$ where:

- G and G_T are groups of order p with efficiently-computable group laws, where p is a λ -bit prime.
- g is a generator of G .
- e is an efficiently-computable bilinear pairing $e : G \times G \rightarrow G_T$, i.e., a map satisfying the following properties:

- Bilinearity: $\forall a, b \in \mathbb{Z}_p, e(g^a, g^b) = e(g, g)^{ab}$;
- Non-degeneracy: $e(g, g) \neq 1$.

Definition 1. *Decisive bilinear division Diffie-Hellman (DBDDH) problem:* Let (p, g, G_1, G_2, e) be the system initial description output. We say the DBDDH assumption holds for description L if the following definition of advantage is negligible in ε :

$$\Pr[L(g, g^a, g^{ab}, e(g, g)^b) = 0] - \Pr[L(g, g^a, g^{ab}, X) = 0] \geq \varepsilon$$

with this probability depending on random selection of a, b, X and output of L .

2.2 A Signature Algorithm

To transform selection plaintext security encryption scheme into chosen-ciphertext security encryption scheme

under standard model, we adopt signature algorithm introduced in [15]. A signature algorithm $Sg = (Gen, Sig, Ver)$ is specified by three polynomial-time algorithms associated with a message space M .

- $Gen(\lambda)$: On input the security parameter λ , this algorithm returns a signature secret key pair $(svk, ssk)/2$.
- $Sig(ssk, M)$: On input public parameter ssk and a message $m \in M$, this algorithm outputs a signature σ .
- If $\sigma = Sig(ssk, M)$, then $Ver(\sigma, svk, M)$ outputs 1, otherwise outputs 0.

In this paper, signature algorithm needs strong unforgeability. That is to say, there is no polynomial-time algorithm attacker for the signed message (M, σ) .

2.3 Proxy Re-encryption

In proxy re-encryption [6, 10] scheme, it allows a partially trusted proxy to transform decrypted ciphertext in Alice as that of Bob. It can guarantee that proxy knows nothing about plaintext [2, 3, 12]. Proxy re-encryption provides effective and safety way for ciphertext conversion, such as digital rights management and mail forwarding. Proxy re-encryption develops very fast in modern time and there are many encryption schemes based on proxy re-encryption to be applied in many aspects.

3 The Model Design

3.1 System Model

The new model is composed of system manager server (SMS), several cloud storage server (CSS), key generation center (KGC) and proxy(P) as figure1. Several users consist of data sharing group. For on user, if he is the data owner (DO), then he will share his information with other users. For DO, other users can share this data that can be called Data Sharer (DS). DO executes the process of secret data encryption and stores the encrypted data in cloud server. SMS would storage some public information in system, such as system public parameters, the user's public key information, users access control. CSS can safely and effectively store the user's sensitive data to ensure the robustness and the integrity of data storage. P transforms encrypted data as ciphertext form which can decrypted by data sharer. Cloud storage data encryption scheme in this paper is specified by five polynomial-time algorithms:

- 1) System initialization. Select a system security parameter. On input this security parameter, this algorithm returns a public parameter. Put this public parameter into SMS, provide access for users and it will be the parameter for user key generation algorithm and data computing operations.

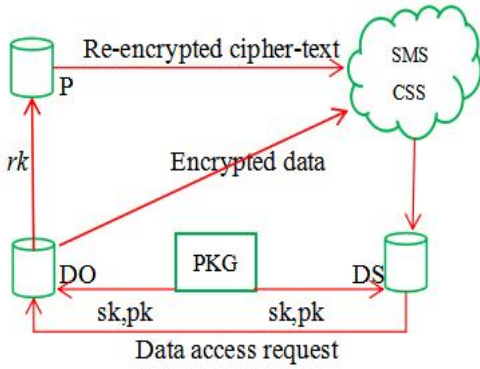


Figure 1: Cloud storage access control system

- 2) Key generation algorithm. KGC can calculate public and private key according to system parameter and user identity information. It will storage the corresponding public key into SMS and provide access for users.
- 3) Data storage algorithm. When Alice wants to share her data with other users. She firstly encrypts plaintext and then puts the encrypted data into CSS. In our scheme, we do not consider robustness and the integrity of data in the cloud storage server. Assuming that CSS can guarantee correctness of the encrypted data. If the correctness of the encrypted data is destroyed, Do must re-encrypt data or re-choose more reliable cloud storage service provider.
- 4) Data re-encrypt algorithm. This algorithm contains re-encryption key generation algorithm and re-encryption algorithm. When Bob wants to get one encrypted data from CSS, he firstly sends data access request to Alice; After receiving the request, Alice visits public key of Bob in SMS and uses re-encryption key generation algorithm to calculate re-encryption key rk . Then Alice sends rk to P . P will adopt re-encryption algorithm to encrypt data and return it back to CSS.
- 5) Data restoration algorithm. Bob again visits CSS to get encrypted data. Using his own private key, he can get restore plaintext information.

3.2 Security Model

Cloud data sharing scheme main aim is to implement data confidentiality and data access control policy. Confidentiality is determined by the encryption algorithm. Access control policy is mainly decided by the re-encryption key generation algorithm. Therefore, we mainly consider the security of data storage and re-encryption key generation algorithm. Security proving is based on cryptography, which is similar to the proxy re-encryption security model. Note that in the following game model, we allow that an

attacker can capture users at any time, namely capturing is flexibility.

Set up the Game. challenger B needs to determine global public parameter for attacker access.

Stage 1. Attacker A can make the any of the following inquiries.

- Public key generation oracle model. A inputs an index i , B inputs a security parameter 1^k . This algorithm gets a pair of public and private key (pk_i, sk_i) . It sends public key pk_i to A , and records (pk_i, sk_i) in table T_K .
- Private key generation oracle model. A inputs a public key pk which is one of output in public oracle model. B finds pk in T_K and returns corresponding sk , public key (pk, pk') .
- Re-encryption key generation oracle model. Attacker inputs (pk, pk') . Challenger returns a re-encryption key $rk_{pk \rightarrow pk'}$ equal to (sk, pk') . sk and pk are public and private key respectively.
- Re-encryption oracle model. A input (pk, pk', C) . B uses sk, pk', C to return a re-encryption ciphertext C' .
- Decryption oracle model. A inputs (pk, C) and B returns decryption result. These inquiries are adaptive. Namely inquiry q_i would be the answer of q_1, \dots, q_{i-1} .

Challenge stage. When Stage 1 is finished. It will output two equi-long plaintext m_0, m_1 and one public key pk^* that will be attacked by A . For pk^* , we cannot use it to inquiry private key and generate oracle. If (pk^*, pk_i) exists in the input of re-encryption key generation oracle model, then pk_i cannot be regarded as input in private key generation oracle model, in that attacker may directly get corresponding plaintext. Challenger randomly select a bit $b \in \{0, 1\}$. Supposing C^* is the ciphertext output after (pk^*, m_b) decrypting.

Stage 2. Attacker can make more inquiries q_{n+1}, \dots, q_n . q_i is one of the following inquiries.

- Public key generation oracle model. Challenger is similar to Stage 1.
- Private key generation oracle model. A inputs a public key $pk, pk \neq pk^*$ which is one of output in public oracle model. (pk^*, pk) is not the output of re-encryption key generation oracle model. (pk^*, pk, C') is not the output of re-encryption oracle model. (pk', C') is subsequent form of (pk^*, C^*) .
- Re-encryption key generation oracle model. Attacker inputs (pk, pk') . pk and pk' is output of public key generation oracle model.

If $pk = pk'$, and pk' is the input of private key oracle model, then B refuses to answer, because the input is an illegal input. Otherwise, it is same to Stage 1.

- Re-encryption oracle model. A input (pk, pk', C) . pk and pk' is output of public key generation oracle model. (pk, C) is subsequent form of (pk^*, C^*) and it is input of private oracle. B refuses to answer, because the input is an illegal input. Otherwise, it is same to Stage 1.
- Decryption oracle model. A inputs (pk, C) . pk is output of public oracle and (pk, C) is not a subsequent form of (pk^*, C^*) . These inquiries are adaptive which is same to Stage 1.

Guessing stage. Finally, attacker outputs one guess $b' \in \{0, 1\}$. If $b = b'$, then attacker wins this game.

The following definition of advantage is negligible under this security model.

$$Adv_{UniIBPRE,A}(k) = [Pr[b = b'] - \frac{1}{2}].$$

4 New Security Cloud Storage Data Encryption Scheme Based on Identity Proxy Re-encryption

4.1 The Detailed Process

We put forward a security cloud storage data encryption scheme based on identity proxy re-encryption and cloud storage. This scheme can realize that users can storage their sensitive data secretly and security share data with other users under the open cloud storage environment. The new scheme is specified by five polynomial-time algorithms: system initialization, key generation, data storage, data re-encryption, data recovery algorithm. Table1 is explanation of symbols used in this paper. Defining $Check$ algorithm: input ciphertext (A, B, C, D, S) and a public key pk . Do the following operations,

- 1) Operating signature algorithm to verify whether signature S is the available signature corresponding to public key svk for algorithm (C, D) .
 - 2) Checking whether equation $e(H_1(sv), B) = e(C, pk)$ is true.
 - 3) If there is one validation failed, then output 0; Otherwise 1.
- System initialization. Input security parameter 1^k , generate system parameter $param$ and main key s .
 - Key generation. Input 1^k , setting $pk = H_{pk}(id)$ and $sk = H_{sk}(id) \cdot s$.

- Data storage. Input pk and plaintext $m \in \{0, 1\}^n$. Do the following operations,

- Choose one signature public-private key pairs $SIG.g(1^k \rightarrow (svk, ssk))$. Setting $Q = svk$.
- Choose a random number $r \in Z_p^*$ and calculate $B = pk', C = H_1(Q)'$, $v = e(g, g)'$, $sk = H(v)$.
- Run symmetric encryption algorithm $SKE.Enc(sk, m)$, m and D are plaintext and ciphertext set respectively.
- Run signature algorithm $SIG.S(SSK, (C, D))$, signed message is (C, D) , having got signature is S .
- Output ciphertext (Q, B, C, D, S) .

- Data re-encryption.

- Re-encryption key generation algorithm. Input a public key pk_2 and a private key sk_1 , output a proxy re-encryption key $rk_{1 \rightarrow 2} = (pk_2)^{\frac{1}{sk_1}}$.
- Re-encryption algorithm. Input re-encryption key $rk_{1 \rightarrow 2}$ and a ciphertext $K = (Q, B, C, D, S)$ encrypted by key pk_1 . If $Check(K, pk_1) = 0$, then output "Reject" and stop. Otherwise operate re-encrypt process to get ciphertext: 1) calculate $B' = e(B, rk_{1 \rightarrow 2})$; 2) Output a new ciphertext $(Q, B, (B', pk_1), C, D, S)$.

- Data recovery. Input a private key sk and a ciphertext K , resolve K . Assuming that $K = (Q, B, C, D, S)$, if $Check(K, g^{sk}) = 0$, then output "Reject" and stop. Otherwise, calculate $v = e(B, g)^{\frac{1}{sk}}$ and $sk = H(v)$. Assuming that $(Q, B, (B', pk_1), C, D, S)$, if $Check(K', pk_1) = 0$ and $K' = (Q, B, C, D, S)$, then output "Reject" and stop. Otherwise, calculate $v = B'^{\frac{1}{sk}}$ and $sk = H(v)$. Then use sk to decrypt $D : SKE.Dec$. Finally, output plaintext m .

4.2 Security Analysis

Theorem 1. *If hypothesis DBDDH is true, our new scheme is CCA security and SIG is strong unforgeable. SKE is security. Especially,*

$$Pr_{B,win} \geq \frac{1}{2} + \frac{1 - (q_{re} + q_d) \cdot \xi}{2e^2(1 + q_{max})} Adv_{PRE,A} - Pr_{AwinSIG} - Pr_{AwinSKE}.$$

A makes q_{re} re-encryption oracle inquiries, q_d decryption oracle inquiries, q_{sk} key generation oracle inquiries at most. $q_{max} = \max\{q_{sk}, q_{rk}, \xi\}$ is verification key maximum probability (supposing it can be ignored) provided by one signature key generation algorithm $SIG.g$. In addition, according to assumption $Pr_{AwinSIG}$ and $Pr_{AwinSKE}$ can be ignored for each probability polynomial time by A .

Table 1: Symbol description

1^k	Security Parameter
$Sig = (G, S, V)$	One Strong Unforgettable Signature Scheme
$SKE = (Enc, Dec)$	A Security Symmetric Encryption Algorithm
q	Prime
G	Groups of Order
g_2, g_3	Two random numbers of Group G_1
$H_1(x)$	$H_1(x) = g_2^x \cdot g_3$
H	$H : G_2 \rightarrow 0, 1^{k_1}$
k_1	Bits Length of Encryption Key
H_{pk}	$H_{pk} : 0, 1^* \rightarrow G$
H_{sk}	$H_{sk} : 0, 1^* \rightarrow Z_q^*$
id	Identity Information

Proof. If there is an attacker A who can break through this scheme, then we build a challenger B to solve $DBDDH$ using algorithm of A . Input (g, g^a, g^{ab}, T) , B judges whether the equation $T = e(g, g)^b$ is true. B sets up the following parameters: bilinear groups $G_1 = (g)$, G_2 , p is bit prime, $e : G_1 \times G_1 \rightarrow G_2$, $(svk^*, ssk^*) \leftarrow g(1^k)$, $A^* = svk^*$, $g_2 = g^{a_1}$, $g_3 = g^{aa_2 - a_1 A^*}$, a_1 and a_2 are two random numbers selected from Z_p^* . Finally, we get $(q, g, g_2, g_3, G_1, G_2, e, H_1, H, SIG, SKE)$.

Challenger B and attacker A do a game according to next procedures. $(A^*, B^*, C^*, D^*, E^*, F^*, S^*)$ denotes no breached challenge ciphertext encrypted by public key pk^* .

Stage 1. B constructs the following oracle model.

- Public key generation oracle. B firstly selects a random number $\varpi \in 0, 1$ for one δ satisfying $Pr[\varpi = 0] = \delta$. B selects one identity information id_i . If $\varpi = 0$, we calculate $pk_i = H_2(id_i)$. Otherwise calculate $pk_i = H_2(id_i) \cdot g^a$. Finally, we record (pk_i, id_i, ϖ_i) into table T_K and return pk_i to attacker. When we input pk_i , B checks whether T_K contains pk_i . If it does not contain, B exits simulation. Otherwise, if $\varpi = 1$, then B reports *failure* and exits; if $\varpi = 0$, B returns $H_3(id_i) \cdot s$ to A and records pk_i into table T_K .
- Re-encryption key oracle. Input (pk_i, pk_j) , B checks whether T_K contains pk_i and pk_j . If it does not contain, B exits simulation. Otherwise, B dose the following operation:
 - If $\varpi_i = \varpi_j$, B uses $g^{\frac{sk_j}{sk_i}}$ to return and records (pk_i, pk_j) into T_K .
 - If $\varpi_i = 0$ and $\varpi_j = 1$, B uses $pk_j \frac{1}{sk_i}$ to return and records (pk_i, pk_j) into T_K .
 - If $\varpi_i = 1$ and $\varpi_j = 0$, then B reports *failure* and exits;
- Re-encryption oracle. Input (pk_i, pk_j, K) , B checks whether T_K contains pk_i and pk_j . If it

does not contain, B exits simulation. Otherwise, if $Check(K, pk_i) = 0$, it shows that the afferent ciphertext is irregular, B outputs *Reject* and exits simulation. Otherwise, B analyzes $K = (Q, B, C, D, S)$ and dose the following operation:

- If $\varpi_i = 1$ and $\varpi_j = 0$, B calculates $t = D/B^{\frac{a_2}{sk_i}}$, $\lambda = \frac{1}{a_1(Q-A^*)}$. Then B can get $B' = e((t^\lambda)^{sk_j}, g)$ and return $(Q, B, (B', pk_i), C, D, S)$ to A . Note when $(Q \neq A^*)$, B can get $t^\lambda = g^r \pi$. In that

$$\begin{aligned}
 t &= \frac{H_1(Q)^r}{(pk_i^r)^{\frac{a_2}{sk_i}}} \\
 &= \frac{g_2^{rA} g_3^r}{pk_i^{ra_2/sk_i}} \\
 &= \frac{(g^{a_1})^{rQ} (g^{aa_2 - a_1 A^*})^r}{(g^{ask_i})^{ra_2/sk_i}} \\
 &= \frac{g^{ra_1(Q-A^*) + ra_2}}{g^{ra_2}} \\
 &= g^{ra_1(Q-A^*)}.
 \end{aligned}$$

Otherwise, B uses (pk_i, pk_j) to inquire re-encryption key oracle and get re-encryption key $rk_{i \rightarrow j}$, then it will execute $ReEnc(rk_{i \rightarrow j}, K)$ and return result to A .

- Decryption oracle. Input (pk_i, K) , B checks whether T_K contains pk_i . If it does not contain, B exits simulation. Otherwise, B dose the following operation:
 - If $\varpi_i = 0$, then $sk_i = H_3(id_i) \cdot s$. B uses $Dec(sk_i, K)$ to return.
 - If $\varpi_i = 1$, then B analyzes K . 1) If $K = (Q, B, C, D, S)$ and $Check(K, pk_i) = 0$, B outputs *Reject* and exits simulation. Otherwise, B gets g^r like in re-encryption oracle and calculates $v = e(g^r, g)$ and $sk = H(v)$.

Then it uses sk to decrypt $D : SKE.Dec$, finally it outputs obtained plaintext m .
 2) If $K = (Q, B, (B', pk_X), C, D, S)$ and $Check(K', pk_X) = 0$, $K' = (Q, B, C, D, S)$, then B outputs *Reject* and exits simulation. Otherwise, B dose the following operation:

- * If $\varpi = 0$, then B calculates $g^r = B \frac{1}{sk_X}$ and checks whether B' is equal to $e(g^r, pk_i)$. If it is false, B outputs *Reject* and exits simulation. Otherwise, B returns $Dec(sk_X, K')$.
- * If $\varpi = 1$, then B is likely in re-encryption oracle getting g^r and checks whether B' is equal to $e(g^r, pk_i)$. If it is false, B outputs *Reject* and exits simulation. Otherwise, B computes $v = e(g^r, g)$ and $sk = H(v)$. Finally, it uses sk to decrypt $D : SKE.Dec$ and outputs obtained plaintext m .

Challenge Stage. Sometime, A can output a challenge tuple (pk^*, m_0, m_1) . If pk^* does not exist in $(T_K$ or $pk^*, pk_i)$ is in T_{rk} and pk_i is in T_{sk} , then B exits simulation. If $\varpi^* = 0$, B reports *failure* and exits simulation. Otherwise, B selects random number $d \in \{0, 1\}$ and calculates:

$$\begin{aligned}
 A^* &= svk^*. \\
 B^* &= (g^{ab})^{sk^*} = (pk^*)^b. \\
 C^* &= (g^{ab})^{a_2} \\
 &= ((g^{a_1}) \cdot g^{aa_2 - a_1 A^*})^b \\
 &= (g_2^{A^*} \cdot g_3)^b \\
 &= H_1(A^*)^b. \\
 v^* &= T. \\
 sk^* &= H(v^*). \\
 D^* &= SKE.Enc(sk^*, m_d). \\
 S^* &= SIG.S(ssk^*, (C^*, D^*)).
 \end{aligned}$$

B returns $K^* = (A^*, B^*, C^*, D^*, S^*)$ to A .

Stage 2. B constructs the following oracle model.

- Public oracle. B resembles Stage 1.
- Private oracle. Input pk_i , if $pk_i = pk^*$ or (pk_i, pk^*) is in T_K , then B exits simulation. Otherwise, it resembles in Stage 1.
- Re-encryption key oracle. Input (pk_i, pk_j) , if $pk_i = pk^*$ and (pk_j) is in T_{sk} , then B exits simulation. Otherwise, it resembles in Stage 1.
- Re-encryption oracle. Input (pk_i, pk_j, K) , if $(pk_i, K) = (pk^*, K^*)$ and (pk_j) is in T_{sk} , then B exits simulation. Otherwise, it resembles in Stage 1.
- Decryption oracle. Input (pk_i, K) , if $(pk_i, K) = (pk^*, K^*)$, then B exits simulation. Otherwise, it resembles in Stage 1.

Guess Stage. At the end, attacker A outputs a guess $d' \in \{0, 1\}$. If $d = d'$, then B outputs 1. Otherwise, it outputs 0.

We firstly analyze *failure* event occurrence rate of ϖ . Its conditions are as follows:

- 1) $\varpi = 0$.
- 2) $\varpi_i = 0$ and $pk_i \neq pk^*$ in private key oracle.
- 3) $\varpi_i = 1$ and $\varpi_j = 0$, $pk_i \neq pk^*$ in re-encryption key oracle.

A makes q_{sk} decryption key oracle inquiries, q_{rk} key generation oracle inquiries at most. So *failure* event occurrence rate of ϖ is $1 - [\delta^{q_{sk}}(1 - (1 - \delta)\delta)^{q_{rk}}]$ in Stages 1, 2. In challenge stage, its occurrence rate is $(1 - \delta)$. Therefore, its total occurrence rate is $1 - [\delta^{q_{sk}}(1 - \delta)(1 - \delta + \delta^2)^{q_{rk}}]$. Now, we assuming that $q_{max} = max\{q_{sk}, q_{rk}\}$, so $\delta^{q_{sk}}(1 - \delta)(1 - \delta + \delta^2)^{q_{rk}} \geq \delta^{q_{max}}(1 - \delta)(1 - \delta + \delta^2)^{q_{max}}$.

When $\delta = \frac{q_{max}}{1 + q_{max}}$, $\delta^{q_{max}}(1 - \delta)$ reaches to maximum value $\frac{1}{e(1 + q_{max})}$, and the rest part $(1 - \delta + \delta^2)^{q_{max}}$ ($q_{max} \rightarrow \infty$) reaches to $\frac{1}{e}$. Therefore,

$$\delta^{q_{max}}(1 - \delta)(1 - \delta + \delta^2)^{q_{max}} \geq \frac{1}{e^2(1 + q_{max})}$$

In addition, when we calculate g' , if $Q = A^* = svk^*$, B will return *failure* that may occur in Stage 1 and Stage 2. Supposing that A makes q_{re} re-encryption oracle inquiries and q_d decryption oracle inquiries, occurrence rate of $Q = A^*$ is $(q_{re} + q_d)\xi$. □

Considering a regular ciphertext (Q, B, C, D, S) can get unique plaintext without encrypted public key. In that $v = e(g, g)^r$, $sk = H(v)$ and $D = SKE.Enc(sk, m)$ uniquely determine plaintext. If $Q = A^* = svk^*$, then the ciphertext is not the subsequent challenge ciphertext. If ciphertext is regular, then S is an effective forged signature of SIG . On the other hand, it can break through SKE , attacker can get d . So we need to minus the probability of A breaking through SKE and SIG from total probability. In this paper, the key size of our method is $n(|U| + |m|) + |G|$, $|U|$ is user's identity length, $|m|$ is message length and $|G|$ is element's length.

5 Conclusions

In this paper, we proposed a new identity proxy re-encryption scheme which was suitable for cloud data sharing. We made a detailed security proving. From the detailed processes, the results illustrated that this new scheme had publicly verification, could filter malicious ciphertext. Meanwhile, it could reach to CCA security standard. In the future, we will improve this encryption scheme and enhance its security to apply it into many actual encryption applications.

Acknowledgments

This study was supported by the Natural Science Foundation of China No.61602080. The authors gratefully acknowledge the anonymous reviewers for their valuable comments.

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, no. 1, pp. 27–38, 2017.
- [2] A. Arriaga, Q. Tang, P. Ryan, "Trapdoor privacy in asymmetric searchable encryption schemes," in *Progress in Cryptology (AFRICACRYPT'14)*, pp. 31–50, 2014.
- [3] D. Biswas, K. Vidyasankar, "Privacy preserving and transactional advertising for mobile services," *Computing*, vol. 96, no. 7, pp. 613–630, 2014.
- [4] J. Han, W. Susilo, Y. Mu, J. Yan, "Privacy-preserving decentralized key-policy attribute-based encryption," *IEEE Transactions on Parallel & Distributed Systems*, vol. 23, no. 11, pp. 2150–2162, 2012.
- [5] W. F. Hsien, C. C. Yang, 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.
- [6] M. M. Jiang, Y. P. Hu, B. C. Wang, Q. Q. Lai, "Lattice-based multi-use unidirectional proxy re-encryption," *Security & Communication Networks*, vol. 8, no. 18, pp. 3796–3803, 2015.
- [7] R. Kangavalli, S. Vagdevi, "A mixed homomorphic encryption scheme for secure data storage in cloud," in *IEEE International Conference on Advance Computing Conference (IACC'15)*, pp. 1062–1066, 2015.
- [8] V. Kgaikwad, R. Kagalkar, "Security and verification of data in multi-cloud storage with provable data possession," *International Journal of Computer Applications*, vol. 117, no. 5, pp. 10–13, 2015.
- [9] J. Lai, R. H. Deng, C. Guan, J. Weng, "Attribute-based encryption with verifiable outsourced decryption," *IEEE Transactions on Information Forensics & Security*, vol. 8, no. 8, pp. 1343–1354, 2013.
- [10] K. Liang, L. Fang, D. S. Wong, W. Susilo, "A ciphertext-policy attribute-based proxy re-encryption scheme for data sharing in public clouds," *Concurrency & Computation Practice & Experience*, vol. 27, no. 8, pp. 2004–2027, 2015.
- [11] L. Liu, J. Ye, "A homomorphic universal re-encryptor for identity-based encryption," *International Journal of Network Security*, vol. 19, no. 1, pp. 11–19, 2017.
- [12] S. Ma, M. Zhang, Q. Huang, B. Yang, "Public key encryption with delegated equality test in a multi-user setting," *Computer Journal*, vol. 58, no. 4, pp. 613–630, 2014.

- [13] M/ Masoumi, M. H. Rezayati, "Novel approach to protect advanced encryption standard algorithm implementation against differential electromagnetic and power analysis," *IEEE Transactions on Information Forensics & Security*, vol. 10, no. 2, pp. 256–265, 2015.
- [14] J. J. Qiu, J. B. Jo, H. J. Lee, "Collusion-resistant identity-based proxy re-encryption without random oracles," *International Journal of Security & Its Applications*, Vol. 9, No. 9, pp. 337–344, 2015.
- [15] C. Ran, S. Halevi, J. Katz, "Chosen-ciphertext security from identity-based encryption," in *International Conference on the Theory and Applications of Cryptographic Techniques*, pp. 207–222, 2004.
- [16] Z. Wang, Y. Lu, G. Sun, "A policy-based deduplication mechanism for securing cloud storage," *International Journal of Electronics and Information Engineering*, vol. 2, no. 2, pp. 70–79, 2015.
- [17] M. Xin, "A mixed encryption algorithm used in internet of things security transmission system," in *IEEE International Conference on Cyber-Enabled Distributed Computing and Knowledge Discovery*, pp. 62–65, 2015.

Biography

Caihui Lan is an associate professor in College of Electronic and Information Engineering, Lanzhou City University. He received his B.S. degree from Northwest University for Nationalities, and received his M.S. and PhD degrees from Northwest Normal University. His research interests include Multimedia Security and Network Security. Email: lanzhourm@163.com.

Haifeng Li is an associate professor in School of Electronic Information and Electrical Engineering, Tianshui Normal University. He received his B.S. and M.S. degrees from Hebei University and Northwest Normal University, respectively. He is currently working toward the PhD degree in School of Software, Dalian University of Technology. His research interests include Multimedia Security, Network Security, and Intelligence Algorithm. Email: lihaifeng8848@mail.dlut.edu.cn.

Shoulin Yin received the B.Eng. And M.Eng. degree from Shenyang Normal University, Shenyang, Liaoning province, China in 2013 and 2015 respectively. His research interests include Multimedia Security, Network Security, Filter Algorithm and Data Mining. He received School Class Scholarship in 2015. Email:352720214@qq.com.

Lin Teng received the B.Eng. degree from Shenyang Normal University, Shenyang, Liaoning province, China in 2016. Now, she is a laboratory assistant in Shenyang Normal University. Her research interests include Multimedia Security, Network Security, Filter Algorithm and Data Mining. Email:1532554069@qq.com.

Policy-based Signatures for Predicates

Fei Tang, Yousheng Zhou

(Corresponding authors: Yousheng Zhou)

College of Cyberspace Security and Law, Chongqing University of Posts and Telecommunications

No. 2 Chongwen Road, Nanan District, Chongqing 400065, China

(Email: zhouys@cqupt.edu.cn)

(Received Nov. 18, 2016; revised and accepted Feb. 20 & Mar. 11, 2017)

Abstract

Policy-based Signatures (PBS), which were introduced by Bellare and Fuchsbaauer, enable signers to sign messages that conform to some policy, yet privacy of the policy is maintained. Bellare et al. defined the policy in any NP language. In PBS schemes for NP language, one should have a valid witness for the policy checking and signing algorithms. In this work, we consider the case of PBS for P language which is a special case of NP language. In PBS schemes for P language, one can directly run the policy checking and signing algorithms without witness. We set policies as some boolean predicates and define the notion of PBS for predicates and its security. Next, for an important class of policy predicates described as (1-dimensional) ranges (i.e., prefix predicate), we design a PBS scheme for such predicate based on tree-based signatures and analyze its application in some real-world scenarios. In addition, based on multilinear maps, we design three PBS schemes for more complex predicates, bit-fixing predicate, left/right predicate, and circuits predicate, respectively.

Keywords: Attribute-based Signatures; Digital Signatures; Group Signatures; Policy-based Signatures

1 Introduction

Digital signatures are one of the most fundamental and well studied cryptographic primitives for providing authentication. In standard signature schemes, a signer who has established a public key pk and a matching secret key sk can sign any message that it wants.

Policy-based Signatures (PBS). The notion of policy-based signatures was introduced by Bellare and Fuchsbaauer [1]. In PBS schemes, signer's secret key sk_p is associated with a policy p that allows the signer to produce a valid signature σ of a message m if and only if the message satisfies the policy. PBS provides flexible and fine-grained privacy-respecting authentication which cannot be provided by the other signature variants. For example, group signatures [6, 14], ring signatures [22, 23, 25],

and attribute-based signatures [20] also have private signing policy. In these signature variants, any verifier can be convinced that the message was signed by someone entitled to, but not who this person is. In addition, in mesh signature scheme [3], the policy itself is always public, as in the warrant, which specifies the policy in proxy signatures [7, 17, 21]. However, note that there has a big difference between policy-based signatures and the other signature variants that the policy-based signatures provide fine-grained control over what kind of messages can be signed by sk_p which associates a policy p .

Bellare et al. [1] defined the policy to be any NP language \mathcal{L} . In order to check that whether a message satisfies a policy or not, they defined a policy checker (i.e., an NP-relation) $PC : \{0,1\}^* \times \{0,1\}^* \rightarrow \{0,1\}$. The first input is a pair (p, m) representing a policy $p \in \{0,1\}^*$ and a message $m \in \{0,1\}^*$, while the second input is a witness $w \in \{0,1\}^*$. The associated language $\mathcal{L}(PC) = \{(p, m) : \exists w \text{ s.t. } m \text{ satisfies } p\}$ is called the policy language associated to PC . Given a witness w , one can test in polynomial time whether a given policy p allows a given message m , where $(p, m) \in \mathcal{L}(PC)$, or $PC((p, m), w) \stackrel{?}{=} 1$ for short.

Our Motivation. Bellare et al. [1] designed a generic construction of PBS scheme for any NP language based on Groth-Sahai proofs [11]. In the scheme of [1], policies can be expressed and enforced are restricted neither in form nor in type, the only condition being that, given a witness, one can test in polynomial time whether a policy allows a message or not. However, in real-world applications, we may only need some specific policies. In addition, it is preferable that check whether a policy allows a message without the help of any witness. In the case of PBS for NP, the witness is necessary for the policy checker and signing algorithm. Hence, in this work, we consider a special case that the policy language in P, meaning that one can directly (without any witness w) run the signing algorithm and test in polynomial time whether a given policy allows a given message, or $p(m) \stackrel{?}{=} 1$ for short.

Our Results. First of all, we define the notion of PBS for predicates, i.e., describe the policies as some boolean predicates, and its security. Then, we design several con-

create PBS schemes for different predicate families.

- Prefix predicates are an important class of policy predicates. Based on tree-based signatures, we design a PBS scheme and prove its security. In addition, we will analyze the applications of PBS scheme for prefix predicates in some real-world scenarios.
- Furthermore, we provide another method to construct PBS for more complex predicates, bit-fixing predicate, left/right predicate, and circuits predicate. The main tool for these three constructions is multilinear map [4]. However, low efficiency which dues to the low efficiency of existing multilinear map candidates and selective unforgeability are two major shortcomings with respect to these multilinear-map-based constructions. Therefore, this part of work is tend to theoretical realization.

2 Preliminaries

2.1 Multilinear Maps

The notion of multilinear maps was introduced by Boneh and Silverberg [4]. Then, Garg et al. [9] gave the first approximate candidate of multilinear maps. Then, many subsequent schemes have been proposed, e.g., [8, 10]. Unfortunately, some of them have been breached, e.g., [12, 18]. However, even so, many cryptographic schemes based on multilinear maps have been proposed, for examples, aggregate signatures [13], attribute-based signatures for circuits [24], constrained PRFs [5] and so on.

Let $\vec{\mathbb{G}} = (\mathbb{G}_1, \dots, \mathbb{G}_k)$ be a sequence of groups each of large prime order p , and g_i be a canonical generator of \mathbb{G}_i , where we set $g = g_1$. There exists bilinear maps $\{e_{i,j} : \mathbb{G}_i \times \mathbb{G}_j \rightarrow \mathbb{G}_{i+j} | i, j \geq 1 \wedge i + j \leq k\}$, which satisfy: $e_{i,j}(g_i^a, g_j^b) = g_{i+j}^{ab} : \forall a, b \in \mathbb{Z}_p$. When the context is obvious, we drop the subscripts i and j , such as $e(g_i^a, g_j^b) = g_{i+j}^{ab}$. It also will be convenient to abbreviate $e(h_1, h_2, \dots, h_j) = e(h_1, e(h_2, \dots, e(h_{j-1}, h_j) \dots)) \in \mathbb{G}_i$ for $h_j \in \mathbb{G}_{i_j}$ and $i_1 + i_2 + \dots + i_j \leq k$. We assume that $\mathcal{G}(1^\lambda, k)$ is a PPT group generator algorithm which takes as input a security parameter λ and a positive integer k to indicate the number of allowed pairing operations, then it outputs the multilinear parameters $mp = (\mathbb{G}_1, \dots, \mathbb{G}_k, p, g = g_1, g_2, \dots, g_k, e_{i,j})$ to satisfy the above properties.

The assumption of Multilinear Computational Diffie-Hellman (MCDH) can be viewed as an adaptation of Bilinear Computational Diffie-Hellman assumption in the setting of multilinear maps.

Definition 1. For any PPT algorithm \mathcal{B} , any polynomial $p(\cdot)$, any integer k , and all sufficiently large $\lambda \in \mathbb{N}$,

$$\Pr \left[\begin{array}{l} mp \leftarrow \mathcal{G}(1^\lambda, k); \\ c_1, \dots, c_k \stackrel{R}{\leftarrow} \mathbb{Z}_p; \\ T \leftarrow \mathcal{B}(mp, g^{c_1}, \dots, g^{c_k}) \end{array} : T = g^{\prod_{i \in [k]} c_i} \right] < \frac{1}{p(\lambda)},$$

where $c_i \stackrel{R}{\leftarrow} \mathbb{Z}_p$ means that c_i is randomly and uniformly chosen from the set \mathbb{Z}_p , and $[k]$ is an abbreviation of the set $\{1, 2, \dots, k\}$.

2.2 Example Predicate Families

We take the predicate families in [5] as examples to realize PBS schemes.

Prefix predicates. Let $v \in \{0, 1\}^n$, where $n \in [k]$, be a bit string, the prefix predicate $p_v : \{0, 1\}^k \rightarrow \{0, 1\}$ is defined as: $p_v(m) = 1 \Leftrightarrow m$ has v as a prefix. The set of prefix predicates is $\mathcal{P}_{pre} = \{p_v : v \in \{0, 1, \perp\}^n, n \in [k]\}$. Prefix predicate is a special case of bit-fixing predicate.

Bit-fixing predicates. Let $\mathbf{v} \in \{0, 1, \perp\}^n$ be a vector, the bit-fixing predicate $p_{\mathbf{v}}^{(BF)} : \{0, 1\}^n \rightarrow \{0, 1\}$ is defined as: $p_{\mathbf{v}}^{(BF)}(m) = 1 \Leftrightarrow (\mathbf{v}_i = m_i \text{ or } \mathbf{v}_i = \perp)$ for all $i = 1, \dots, n$. The set of bit-fixing predicates is $\mathcal{P}_{BF} = \{p_{\mathbf{v}}^{(BF)} : \mathbf{v} \in \{0, 1, \perp\}^n\}$.

Left/right predicates. For a bit string $w \in \{0, 1\}^{|m|/2}$, where $m \in \mathcal{M}$ and $|m| = 2 \cdot s$ denotes the size of the message, define two predicates $p_w^{(L)}, p_w^{(R)} : \{0, 1\}^{|m|/2} \rightarrow \{0, 1\}$ as: $p_w^{(L)}(m_1, m_2) = 1 \Leftrightarrow m_1 = w$ and $p_w^{(R)}(m_1, m_2) = 1 \Leftrightarrow m_2 = w$.

Circuit predicates. Let \mathcal{C} be the set of polynomial size circuits. Circuit predicate family is defined as $\mathcal{P}_{cir} = \{p : p \in \mathcal{C}\}$.

3 Policy-based Signatures for Predicates

3.1 Definition

A policy-based signature scheme, PBS , consists of the following four PPT algorithms:

- **Setup**(1^λ): The setup algorithm takes as input a security parameter λ . It outputs public parameters pp and a master secret key msk , where the public parameters, pp , contain the descriptions of the message space \mathcal{M} , signature space \mathcal{S} , and policy (boolean predicate) space \mathcal{P} . The master secret key msk can sign all messages in \mathcal{M} .
- **KeyGen**(msk, p): The key generation algorithm takes as input the master secret key msk and a boolean predicate $p \in \mathcal{P}$. It outputs a signing key sk_p for the predicate p .
- **Sign**(sk_p, m): The signing algorithm takes as input a signing key sk_p and a message $m \in \mathcal{M}$. It outputs a signature $\sigma \in \mathcal{S}$ if $p(m) = 1$. Otherwise, it outputs \perp .

- **Verify**(pp, m, σ): The verification algorithm takes as input the public parameters pp and a purported signature σ for a message m . It outputs 1 or 0.

We require that for all security parameter λ , $(msk, pp) \leftarrow \text{Setup}(1^\lambda)$, $p \in \mathcal{P}$, $sk_p \leftarrow \text{KeyGen}(msk, p)$, and $m \in \mathcal{M}$, if $p(m) = 1$ and $\sigma \leftarrow \text{Sign}(sk_p, m)$, then we have $\text{Verify}(pp, m, \sigma) = 1$.

3.2 Security Models

The security of policy-based signature for predicates is defined by the following two notions: unforgeability and privacy (i.e., indistinguishability in [1]).

Unforgeability. This notion guarantees that one can sign some message m only if it has a signing key sk_p such that $p(m) = 1$.

- **Setup:** The challenger runs the setup algorithm to generate public parameters pp and a master secret key msk . It then gives pp to the adversary and keeps msk to itself.
- **Key Generation Oracle:** The adversary adaptively makes any polynomial number of signing key queries for boolean predicate $p \in \mathcal{P}$ of its choice. The challenger returns back $sk_p \leftarrow \text{KeyGen}(msk, p)$.
- **Signing Oracle:** The adversary adaptively makes any polynomial number of signature queries on input a message $m \in \mathcal{M}$. The challenger chooses a $p \in \mathcal{P}$ such that $p(m) = 1$ and returns back $\sigma \leftarrow \text{Sign}(sk_p, m)$, where sk_p is obtained from the key generation algorithm.
- **Forgery:** The adversary finally outputs a tuple (m^*, σ^*) . It wins the game if (1) $\text{Verify}(pp, m^*, \sigma^*) = 1$; (2) m^* was never queried to the signing oracle; and (3) $p(m^*) = 0$ for all p queried to the key generation oracle.

We denote the advantage of a PPT adversary \mathcal{A} (taken over the random choices of the challenger and adversary) to win the game as $\text{Adv}_{\mathcal{A}}^{\text{Unf}} = \Pr[\mathcal{A} \text{ wins}]$.

Definition 2. A policy-based signature scheme is existentially unforgeable if any PPT adversary can win the above game with at most negligible advantage.

Remark 1. In the case of PBS for NP, the above notion is unsatisfactory. This is because one cannot efficiently verify whether an adversary has won the game, as this needs it has a valid witness w to checking that whether $(p, m) \in \mathcal{L}(\text{PC})$ for all p queried to the key generation oracle and m from the adversary's final output. However, in the case of PBS for P, one can always efficiently verify it without any witness.

We also define a weaker (selective) variant to the above definition where the adversary is required to commit to a challenge message, m^* , before the setup phase.

Definition 3. A policy-based signature scheme is selectively unforgeable if any PPT adversary can win the selective game with at most negligible advantage.

Perfect Privacy. This notion guarantees that a valid signature will reveal nothing about the signing policy.

- **Setup:** The challenger runs the setup algorithm to generate public parameters pp and a master secret key msk . It then gives pp and msk to the adversary.
- **Challenge:** The adversary submits a challenge message $m^* \in \mathcal{M}$ and two different boolean predicates $p_0, p_1 \in \mathcal{P}$ such that $p_0(m^*) = p_1(m^*) = 1$, to the challenger. The challenger flips a random coin $b \leftarrow \{0, 1\}$ and returns back $\sigma_b \leftarrow \text{Sign}(sk_{p_b}, m^*)$, where sk_{p_b} is obtained from the key generation algorithm.
- **Guess:** Finally, the adversary outputs his guessing bit b' and wins the game if $b' = b$.

We denote the advantage of an unbounded adversary \mathcal{A} (taken over the random choices of the challenger and adversary) to win the game as $\text{Adv}_{\mathcal{A}}^{\text{Pri}} = |\Pr[b' = b] - \frac{1}{2}|$.

Definition 4. A policy-based signature scheme is perfectly private if even an unbounded adversary wins the above game with at most negligible advantage.

4 Policy-based Signatures Based on One-Way Functions

We now construct a policy-based signature scheme supporting the class of prefix predicate based on the tree-based signature scheme [19].

4.1 Tree-based Signature Scheme

Let $\mathcal{TTS} = (\text{keygen}, \text{sign}, \text{verify})$ be a two-time signature scheme.¹ For a binary string m , let $m|_i = m_1 \cdots m_i$ denote the i -bit prefix of m (with $m|_0 := \varepsilon$, the empty string). More specifically, we imagine a binary tree of depth k where the root is labelled by ε (i.e., the empty string), and a node that is labelled with a binary string w , where $|w| < k$, has left-child labelled $w0$ and right-child labelled $w1$. For every node w , we associate a key pair (pk_w, sk_w) from the two-time signature scheme. The public key of the root, pk_ε , is the actual public key of the signer. To sign a message $m \in \{0, 1\}^k$, the signer does the following steps:

- 1) It first generates keys for all nodes on the path from the root to the leaf labelled m . Some of these public keys may generated in the process of signing previous

¹The original tree-based signature scheme is based on any one-time signature (e.g., [16]). For ease of description, we define the tree-based signature scheme based on any \mathcal{TTS} which can be easily realized from any one-time signature scheme.

messages, in such case the previous values are stored as part of the state.

- 2) It then “certifies” the path from the root to the leaf labelled m by computing a signature on pk_{w0} or pk_{w1} (which depends on whether $m_{|w|+1} = 0$ or 1), using secret key sk_w , for each string w that is a proper prefix of m .
- 3) Finally, the signer “certifies” m itself by computing a signature on m using the secret key sk_m .

Formally, the tree-based signature scheme $\mathcal{TBS} = (\text{keygen}^*, \text{sign}^*, \text{verify}^*)$ is as follows:

keygen $^*(1^\lambda)$: The key generation algorithm runs $(pk_\varepsilon, sk_\varepsilon) \leftarrow \text{keygen}(1^\lambda)$ and outputs the public key pk_ε . The secret key and initial state are sk_ε .

sign $^*(sk, m \in \{0, 1\}^k)$: To sign a message $m \in \{0, 1\}^k$ using the current state, the signing algorithm does the following:

- 1) For $i = 0, \dots, k - 1$: let $m_{|i}b$ be the $(i + 1)$ -bit prefix of m .
 - If $pk_{m_{|i}b}$ and $\sigma_{m_{|i}b}$ are not in the current state, compute them:

$$(pk_{m_{|i}b}, sk_{m_{|i}b}) \leftarrow \text{keygen}(1^\lambda);$$

$$\sigma_{m_{|i}b} \leftarrow \text{sign}(sk_{m_{|i}b}, pk_{m_{|i}b}),$$
 and then store all these computed values as part of the state.
- 2) If σ_m is not yet included in the state, compute $\sigma_m \leftarrow \text{sign}(sk_m, m)$ and store it as part of the state.
- 3) Output $\sigma = (\{(pk_{m_{|i}b}, \sigma_{m_{|i}b})\}_{i \in [0, k-1]}, \sigma_m)$ as the signature for message m .

verify $^*(pk, m, \sigma)$: Given a public key pk_ε , a message $m \in \{0, 1\}^k$, and a signature $(\{(pk_{m_{|i}b}, \sigma_{m_{|i}b})\}_{i=0}^{k-1}, \sigma_m)$, output 1 if and only if the following two equations hold:

- 1) $\text{verify}(pk_{m_{|i}}, pk_{m_{|i}b}, \sigma_{m_{|i}b}) = 1$ for all $i = 0, \dots, k - 1$.
- 2) $\text{verify}(pk_m, m, \sigma_m) = 1$.

Theorem 1. *If \mathcal{TTS} is a signature scheme that is existentially unforgeable against two-time chosen-message attack. Then the tree-based signature scheme \mathcal{TBS} is existentially unforgeable against adaptive chosen-message attack.*

The proof this theorem is similar to that of the tree-based signature scheme based on any one-time signature scheme. We omit the proof.

4.2 PBS for Prefix Predicates based on \mathcal{TBS}

We now construct a policy-based signature scheme for the prefix predicates based on the tree-based signature scheme. The idea of our construction is as follows:² a signing key sk_{p_v} corresponding to a predicate $p_v \in \mathcal{P}_{pre}$ will be the partial certification of the \mathcal{TBS} tree, at level $|v|$. Given this partial certification, a signer will be able to compute the completion for any message m which has v as a prefix. However, as we will argue, the computation of all other messages will remain unknown to the signer. Formally, our PBS scheme (**Setup**, **KeyGen**, **Sign**, **Verify**) for prefix predicates is as follows:

Setup (1^λ) : This algorithm runs $(pk_\varepsilon, sk_\varepsilon) \leftarrow \text{keygen}^*(1^\lambda)$ and sets $msk := sk_\varepsilon, pp := pk_\varepsilon$.

KeyGen (msk, p_v) : To compute a signing key sk_{p_v} for a prefix predicate $p_v \in \mathcal{P}_{pre}$, where $|v| \leq k$. This key generation algorithm does the following:

- 1) For $i = 0, \dots, |v| - 1$: let $v_{|i}b$ be the $(i + 1)$ -bit prefix of the vector v .
 - If $pk_{v_{|i}b}$ and $\sigma_{v_{|i}b}$ are not in the current state, compute them:

$$(pk_{v_{|i}b}, sk_{v_{|i}b}) \leftarrow \text{keygen}(1^\lambda);$$

$$\sigma_{v_{|i}b} \leftarrow \text{sign}(sk_{v_{|i}b}, pk_{v_{|i}b}),$$
 and then store all these computed values as part of the state.
- 2) Output $sk_{p_v} = (\{(pk_{v_{|i}b}, \sigma_{v_{|i}b})\}_{i \in [0, |v|-1]}, sk_v)$ as the signing key for p_v .

Sign $(sk_{p_v}, m \in \{0, 1\}^k)$: To sign a message $m \in \{0, 1\}^k$ using the current state, the signing algorithm does the following:

- 1) If $p_v(m) = 0$, i.e., $\exists i \in [|v|]$ s.t. $m_i \neq v_i$, then abort.
- 2) For $i = |v|, \dots, k - 1$: let $m_{|i}b$ be the $i + 1$ -bit prefix of m .
 - If $pk_{m_{|i}b}$ and $\sigma_{m_{|i}b}$ are not in the current state, compute them:

$$(pk_{m_{|i}b}, sk_{m_{|i}b}) \leftarrow \text{keygen}(1^\lambda);$$

$$\sigma_{m_{|i}b} \leftarrow \text{sign}(sk_{m_{|i}b}, pk_{m_{|i}b}),$$
 and then store all these computed values as part of the state.
- 3) If σ_m is not yet included in the state, compute $\sigma_m \leftarrow \text{sign}(sk_m, m)$ and store it as part of the state.
- 4) Output $\sigma = (\{(pk_{m_{|i}b}, \sigma_{m_{|i}b})\}_{i \in [0, k-1]}, \sigma_m)$ as the signature for message m .

Verify (pp, m, σ) : It is same the verify^* algorithm in the \mathcal{TBS} scheme.

²A similar idea has been used to construct constrained PRFs [2].

Unforgeability. Our PBS scheme \mathcal{PBS} is similar to the tree-based signature scheme \mathcal{TBS} . The only difference is that in \mathcal{PBS} the signer obtains a signing key sk_{p_v} , rather than a full-featured key $msk := sk_\varepsilon$ as in \mathcal{TBS} , which enables him to signing a subset of the domain $\{0, 1\}^k$. For completeness, we give the following proof.

Theorem 2. *If \mathcal{TTS} is a signature scheme that is existentially unforgeable under a two-time chosen-message attack. Then the policy-based signature scheme \mathcal{PBS} for message space $\{0, 1\}^k$ is existentially unforgeable under an adaptive chosen-message attack.*

Proof. For length of messages k , we prove security based on two-time signature scheme \mathcal{TTS} . We show that if there exists a PPT adversary \mathcal{A} on our PBS scheme then we can construct an efficient algorithm \mathcal{B} to break the security of scheme \mathcal{TTS} . We describe how \mathcal{B} interacts with \mathcal{A} . The algorithm \mathcal{B} first receives a challenge key pk from the challenger of scheme \mathcal{TTS} .

Let q_k, q_s be the upper bounds on the number of key generation queries and signing queries, respectively, made by \mathcal{A} , and set $\ell = k(q_k + q_s) + 1$. Note that ℓ upper-bounds the number of public keys from \mathcal{TTS} that are needed to generate q_k keys and q_s signatures using \mathcal{PBS} (in the worst case), and there is one additional key form \mathcal{TTS} that is used as the actual public key pk_ε .

Setup: Initially, \mathcal{B} chooses a random $i^* \leftarrow [\ell]$, we assume that i^* is put on the node (or leaf) w^* . Construct a list pk^1, \dots, pk^ℓ of keys as follows:

- Set $pk^{i^*} := pk$.
- For $i \neq i^*$, run $(pk^i, sk^i) \leftarrow \text{keygen}(1^\lambda)$.

Then \mathcal{B} runs \mathcal{A} on input the public key $pk_\varepsilon := pk^1$. Note that \mathcal{B} knows all secret key sk^i , for $i \in [\ell]$, except that the challenge one $sk^{i^*} := sk$. With respect to the challenge key sk , \mathcal{B} can make at most two signing queries to the \mathcal{TTS} challenger according to the security of the \mathcal{TTS} scheme.

Key Generation Oracle: \mathcal{A} will query for a signing key for a prefix predicate $p_v \in \mathcal{P}_{pre}$. \mathcal{B} creates it for the adversary as follows:

- If w^* is not on the path from the root to the node v , then \mathcal{B} can create signing key sk_{p_v} honestly since it knows all secret keys with respect to the nodes on the path from the root to the node v .
- If w^* is on the path from the root to the node v , to certify w^* 's child w^*b , \mathcal{B} first makes a signing query to the \mathcal{TTS} challenger on input pk^{i^*+1} , and then it will receive a signature $\sigma_{w^*b} \leftarrow \text{sign}(sk, pk^{i^*+1})$. Finally, \mathcal{B} can create sk_{p_v} .

Signing Oracle: The adversary \mathcal{A} will query for a signature for a message $m \in \{0, 1\}^k$. \mathcal{B} creates signatures for the adversary as follows:

- If w^* is not on the path from the root to the leaf m , then \mathcal{B} can create signature σ honestly since it knows all secret keys with respect to the nodes on the path from the root to the leaf m .
- If w^* is on the path from the root to the leaf m , and if w^* is an internal node, to certify the w^* 's child w^*b , \mathcal{B} first makes a signing query to the \mathcal{TTS} challenger on input pk^{i^*+1} and then it will receive a signature $\sigma_{w^*b} \leftarrow \text{sign}(sk, pk^{i^*+1})$; if w^* is the leaf m , then \mathcal{B} makes signing query to the \mathcal{TTS} challenger on input m and then it will receive a signature $\sigma_m \leftarrow \text{sign}(sk, m)$. Finally \mathcal{B} can compute the complete signature σ .

Forgery: Eventually, \mathcal{A} outputs

$$\sigma^* = (\{(pk'_{m^*|ib}, \sigma'_{m^*|ib})\}_{i=0}^{k-1}, \sigma'_{m^*})$$

for message m^* . If it is valid, then:

Case 1: There exists a $j \in [0, k-1]$ for which $pk'_{m^*|ib} \neq pk_{m^*|ib}$, this means that \mathcal{A} creates a new key $pk'_{m^*|ib}$ but not initially defined by \mathcal{B} . If $j = i^*$, \mathcal{B} outputs $(pk'_{m^*|ib}, \sigma'_{m^*|ib})$.

Case 2: If Case 1 does not hold, then $pk'_{m^*} = pk_{m^*}$. Let j be such that $pk^j = pk_{m^*}$. If $j = i^*$, \mathcal{B} outputs (m^*, σ'_{m^*}) .

Note that i^* was chosen uniformly at random and is independent of the view of \mathcal{A} , and the list pk^1, \dots, pk^ℓ generated by $\text{keygen}(1^\lambda)$ is distributed identically to the view of \mathcal{A} in the real unforgeability game. Thus if \mathcal{A} outputs a valid forgery (regardless of which of the above cases occurs) with probability ϵ , \mathcal{B} can outputs a forgery with probability ϵ/ℓ , where $1/\ell$ means the probability of $j = i^*$ in which of the above cases occurs. By the assumed security of \mathcal{TTS} and the fact that ℓ is polynomial, we conclude that ϵ must be negligible. \square

Perfect Privacy. Given a valid signature (m^*, σ^*) , we show that any key sk_{p_v} such that $p_v(m^*) = 1$ could possibly have created it. The proof is straightforward.

Theorem 3. *The above PBS scheme based on OWF is perfectly private.*

Proof. Any unbounded adversary \mathcal{A} submits a challenge tuple (p_v, p_w, m^*) such that $p_v(m^*) = p_w(m^*) = 1$. The distribution of the signatures for m^* generated by the signing key sk_{p_v} is: $(\{(pk'_{m^*|ib}, \sigma'_{m^*|ib})\}_{i \in [0, k-1]}, \sigma_{m^*})$, where each key pair (pk_i, sk_i) is generated by the key generation algorithm randomly and independently. Similarly, The distribution of the signatures for m^* generated by the signing key sk_{p_w} is: $(\{(pk'_{m^*|ib}, \sigma'_{m^*|ib})\}_{i \in [0, k-1]}, \sigma'_{m^*})$,

where each key pair (pk'_i, sk'_i) also is generated by the key generation algorithm randomly and independently. Therefore, these two distributions are identical. The perfect privacy follows easily from this observation. \square

4.3 Application of PBS for Prefix Predicates

We consider the application scenario which was raised by Bellare et al. [1]. A company implements a scheme where each employee gets a signing key with a policy and there is only one public key which is used by outsiders to verify signatures in the name of the company. Company stipulates that employee in different department has different policy. For example, the policy for sales department states the prices of the products, the policy for technology department states the functionalities of the products, and so on.

PBS for prefix predicates is a useful tool in addressing this problem. For an employee in sales department, company sets policy p_s which states the prices of the products, then distributes a signing key sk_{p_s} to this employee. Finally, the employee can sign messages on behalf of the company, where these messages may contain the statement “product prices||after-sale service terms||...”. In such a scenario, the employee can decide the after-sale service terms and some other regulations, however, the product prices which are stipulated in the policy p_s cannot be changed. If the PBS scheme for prefix predicates is secure, then any outsider can be convinced that, from a valid PBS signature, the regulations (i.e., message) was agreed by someone entitled to, but not who this person is.

Related Works. Append-Only Signatures (AOS) [15] are a similar notion to the PBS for prefix predicates. In AOS schemes, any party is given an AOS signature σ_{m_1} for message m_1 can compute $\sigma_{m_1||m_2}$ for message $m_1||m_2$, where the message m_2 is chosen by the party. In AOS, *anyone* can append and verify signatures. However, in PBS, the signing key sk_p cannot be opened, and hence only the holder of the signing key can sign messages. Kiltz et al. [15] showed that AOS is equivalent to Hierarchical Identity-Based signatures (HIBS), and it can be used to the Border Gateway Protocol (BGP). PBS for prefix predicates and AOS are two different signature variants because: (1) there has no obvious evidence shows that the PBS for prefix predicates has the properties (i.e., connection to HIBS and application to BGP) provided by the AOS; and (2) AOS apparently does not apply to the above application scenario because, in the above application scenario, the signer should to be some authorized employee rather than anyone. Although, there may be have some potential connections between these two signature variants, e.g., realize one from the other one by some transformation, which beyond the reach of this work.

5 Policy-based Signatures Based on Multilinear Maps

In this section, we take advantage of the multilinear maps to realize three PBS constructions for bit-fixing predicates, left/right predicates, and circuit predicates, respectively. The main technique of our multilinear-map-based PBS schemes follows Boneh and Waters’ [5] work which constructs constrained pseudorandom functions. Their technique has been used to construct different cryptographic primitives, such as attribute-based signatures for circuits [24] and so on. In this work, we also follow Boneh et al.’s [5] technique, however, to construct a different cryptographic primitive, policy-based signatures.

5.1 PBS for Bit-Fixing Predicates

Setup $(1^\lambda, k)$: The setup algorithm takes as input a security parameter λ and an integer k . The algorithm then runs $\mathcal{G}(1^\lambda, k)$ that produces groups $\overline{\mathbb{G}} = (\mathbb{G}_1, \dots, \mathbb{G}_k)$ of prime order p , with canonical generators g_1, \dots, g_k , where we let $g = g_1$. Next it chooses random $\alpha \in \mathbb{Z}_p$ and $(a_{1,0}, a_{1,1}), \dots, (a_{k-1,0}, a_{k-1,1}) \in \mathbb{Z}_p^2$ and computes $A_{i,\beta} = g^{a_{i,\beta}}$ for $i \in [k-1]$ and $\beta \in \{0, 1\}$.

The master secret key is $msk = \alpha$. The public parameters, pp , consist of the group sequence description plus group elements $g^\alpha, \{A_{i,\beta} | i \in [k-1], \beta \in \{0, 1\}\}$, message space $\mathcal{M} = \{0, 1\}^{k-1}$, signature space $\mathcal{S} = \mathbb{G}_{k-1}$, and predicate space $\mathcal{P} = \{0, 1, \perp\}^{k-1}$.

KeyGen $(msk, \mathbf{v} \in \{0, 1, \perp\}^{k-1})$: For a vector $\mathbf{v} \in \{0, 1, \perp\}^{k-1}$, let V be the set of indices $i \in [k-1]$ such that $\mathbf{v}_i \neq \perp$. That is the indices for which the bit is fixed to 0 or 1. The signing key for the predicate $p_{\mathbf{v}}$ is:

$$sk_{\mathbf{v}} = (g_{|V|}^{\prod_{i \in V} a_{i, \mathbf{v}_i}})^\alpha \in \mathbb{G}_{|V|}.$$

Sign $(sk_{\mathbf{v}}, m \in \{0, 1\}^k)$: Given a message, m , of length $k-1$, let m_1, \dots, m_{k-1} be the bits of this message. If the message m does not satisfy the predicate, i.e., $p_{\mathbf{v}}(m) = 0$, then abort. Otherwise, $p_{\mathbf{v}}(m) = 1$. That is $\mathbf{v}_i = m_i$ for all $i \in V$, then the signing algorithm compute a signature:

$$\sigma = e(sk_{\mathbf{v}}, g_{k-|V|-1}^{\prod_{i \in [k-1] \setminus V} a_{i, m_i}}) = g_{k-1}^{\alpha \cdot \prod_{i \in [k-1] \setminus V} a_{i, m_i}} \in \mathbb{G}_{k-1},$$

where $g_{k-|V|-1}^{\prod_{i \in [k-1] \setminus V} a_{i, m_i}}$ can be computed by the multilinear maps from the parameters A_{i, m_i} for $i \in [k-1] \setminus V$.

Verify (pp, m, σ) : Given a purported signature σ on a message m , verify the following equation:

$$e(\sigma, g) = e(g^\alpha, A_{1, m_1}, \dots, A_{k-1, m_{k-1}}).$$

Output 1 if it holds, else 0.

Correctness. The verification of the final signatures is justified by the following two equations:

$$\begin{aligned} e(\sigma, g) &= e\left(\left(g_{k-1}^{\prod_{i \in [k-1]} a_{i, m_i}}\right)^\alpha, g\right) \\ &= g_k^{\alpha \cdot \prod_{i \in [k-1]} a_{i, m_i}}. \end{aligned}$$

and

$$e(g^\alpha, A_{1, m_1}, \dots, A_{k-1, m_{k-1}}) = g_k^{\alpha \cdot \prod_{i \in [k-1]} a_{i, m_i}}.$$

Theorem 4. *If the k -MCDH assumption is hold in the multilinear groups, then the above PBS construction for bit-fixing predicates and for messages of length $k - 1$ is selectively unforgeable and perfectly private.*

Selective Unforgeability. We prove selective unforgeability, where the key access structures are bit-fixing predicates. For length of messages $k - 1$, we prove security under the k -MCDH assumption. We show that if there exists a PPT adversary \mathcal{A} on our PBS scheme for messages of length $k - 1$ in the selective security game then we can construct an efficient algorithm \mathcal{B} on the k -MCDH assumption. We describe how \mathcal{B} interacts with \mathcal{A} .

The algorithm \mathcal{B} first receives a k -MCDH challenge instance consisting of the group sequence description \mathbb{G} and $g = g_1, g^{c_1}, \dots, g^{c_k}$. It also receives challenge message $m^* = m_1^* \dots m_{k-1}^* \in \{0, 1\}^{k-1}$ from the adversary.

Setup: Initially, \mathcal{B} chooses random $u_1, \dots, u_{k-1} \in \mathbb{Z}_p$ and sets

$$A_{i, \beta} = \begin{cases} g^{c_i}, & \text{if } m_i^* = \beta \\ g^{u_i}, & \text{if } m_i^* \neq \beta \end{cases}$$

for $i \in [k-1], \beta \in \{0, 1\}$. This corresponds to setting $a_{i, \beta} = c_i$ if $m_i^* = \beta$ and u_i otherwise. We observe these are distributed identically to the real scheme. In addition, it will internally view $\alpha = c_k$.

Key Generation Oracle: The adversary \mathcal{A} will query for a signing key for a bit-fixing predicate $p_{\mathbf{v}}$, where $\mathbf{v} \in \{0, 1, \perp\}^{k-1}$. We let V be the set of indices $i \in [k-1]$ such that $\mathbf{v}_i \neq \perp$.

If $p_{\mathbf{v}}(m^*) = 1$, that is $m_i^* = \mathbf{v}_i$ for all $i \in V$. Then \mathcal{B} aborts the game.

If $p_{\mathbf{v}}(m^*) = 0$, that is $\exists j \in [k-1] \setminus V$, s.t. $m_j^* \neq \mathbf{v}_j$. Then \mathcal{B} will be able to create signing keys for the adversary, because his query will differ from the challenge message at least one bit. More specifically, \mathcal{B} produces the signing key as $sk_{\mathbf{v}} = e(g^{c_k}, g_{|V|-1}^{\prod_{i \neq j \in V} a_{i, x_i}}) u_j$.

Signing Oracle: The adversary \mathcal{A} will query for a signature for a message $m \neq m^*$, and we let $m_j \neq m_j^*, j \in [k-1]$. Then \mathcal{B} will be able to produce a valid signature:

$$\sigma = e(g^{c_k}, g_{k-2}^{\prod_{i \neq j \in [k-1]} a_{i, m_i}}) u_j.$$

Forgery: Eventually, \mathcal{A} outputs a signature σ^* on message m^* . Then \mathcal{B} outputs σ^* as the solution of the given instance of the k -MCDH assumption.

According to the public parameters built in the setup phase and the assumption that σ^* is valid, we know that $\sigma^* = g_{k-1}^{\prod_{i \in [k]} c_i}$, implies that σ^* is a solution for the given instance of the k -MCDH problem, and thus \mathcal{B} breaks the k -MCDH assumption. It is clear that the view of \mathcal{A} simulated by \mathcal{B} in the above game is distributed statistically exponentially closely to that in the real unforgeability game, hence \mathcal{B} succeeds whenever \mathcal{A} does. \square

Perfect Privacy. Given a valid signature (m^*, σ^*) , we show that any signing key $sk_{\mathbf{v}}$ such that $p_{\mathbf{v}}(m^*) = 1$ could possibly have created it. The proof is straightforward.

According to the setup of the signing algorithm, for any tuple $(\mathbf{v}[0], \mathbf{v}[1], m^*)$ such that $p_{\mathbf{v}[0]}(m^*) = p_{\mathbf{v}[1]}(m^*) = 1$, which was chosen by an unbounded adversary \mathcal{A} , both of the signatures created by the signing key $sk_{\mathbf{v}[0]}$ and $sk_{\mathbf{v}[1]}$ are $g_{k-1}^{\alpha \cdot \prod_{i \in [k-1]} a_{i, m_i^*}}$. Therefore, any signing key $sk_{\mathbf{v}}$ such that $p_{\mathbf{v}}(m^*) = 1$ can compute a same signature on a given message m^* . The perfect privacy follows easily from this observation. \square

5.2 PBS for Left/Right Predicates

Setup($1^\lambda, k = 2 \cdot s + 1$): The setup algorithm takes as input a security parameter λ and an odd number $k = 2 \cdot s + 1$. The algorithm then runs $\mathcal{G}(1^\lambda, k)$ that produces groups $\mathbb{G} = (\mathbb{G}_1, \dots, \mathbb{G}_k)$ of prime order p , with canonical generators g_1, \dots, g_k , where we let $g = g_1$. Next it chooses random $\alpha \in \mathbb{Z}_p, (a_{1,0}, a_{1,1}), \dots, (a_{s,0}, a_{s,1}), (b_{1,0}, b_{1,1}), \dots, (b_{s,0}, b_{s,1}) \in \mathbb{Z}_p^2$ and computes $A_{i, \beta} = g^{a_{i, \beta}}, B_{i, \beta} = g^{b_{i, \beta}}$ for $i \in [s]$ and $\beta \in \{0, 1\}$.

The master secret key is $msk = \alpha$. The public parameters, pp , consist of the group sequence description plus group elements $g^\alpha, \{A_{i, \beta}, B_{i, \beta} | i \in [s], \beta \in \{0, 1\}\}$, message space $\mathcal{M} = \{0, 1\}^{k-1}$, signature space $\mathcal{S} = \mathbb{G}_{k-1}$, and predicate space $\mathcal{P} = \{0, 1\}^s$.

KeyGen($msk, p = (p_x^{(L)}, p_y^{(R)})$): For $(x, y) \in \{0, 1\}^s$, the signing keys for the predicates $p_x^{(L)}, p_y^{(R)}$ are $sk_{p_x^{(L)}} = (g_s^{\prod_{i \in [s]} a_{i, x_i}})^\alpha \in \mathbb{G}_s$ and $sk_{p_y^{(R)}} = (g_s^{\prod_{i \in [s]} b_{i, y_i}})^\alpha \in \mathbb{G}_s$, respectively.

Sign(sk_p, m): Given a message, m , of length $k - 1$, let $m_1, \dots, m_s, m_{s+1}, \dots, m_{k-1}$ be the bits of this message. If the message m satisfies the left predicate $p_x^{(L)}$, i.e., $m_i = x_i$ for $i \in [s]$, then the signing algorithm can compute a signature:

$$\begin{aligned} \sigma &= e(sk_{p_x^{(L)}}, B_{1, m_{s+1}}, \dots, B_{s, m_{k-1}}) = \\ &= g_{k-1}^{\alpha \cdot \prod_{i \in [s]} a_{i, m_i} \cdot \prod_{i \in [s]} b_{i, m_{s+i}}} \in \mathbb{G}_{k-1}. \end{aligned}$$

If the message m satisfies the right predicate $p_y^{(R)}$, i.e., $m_{s+i} = y_i$ for $i \in [s]$, then the signing algorithm can compute a signature:

$$\sigma = e(sk_{p_y^{(R)}}, A_{1,m_1}, \dots, A_{s,m_s}) = g_{k-1}^{\alpha \cdot \prod_{i \in [s]} a_{i,m_i} \cdot \prod_{i \in [s]} b_{i,m_{s+i}}} \in \mathbb{G}_{k-1}.$$

Verify(pp, m, σ): Given a purported signature σ on a message m , verify the following equation:

$$e(\sigma, g) = e(g^\alpha, A_{1,m_1}, \dots, A_{s,m_s}, B_{1,m_{s+1}}, \dots, B_{s,m_{k-1}}).$$

Output 1 if it holds, else 0.

Correctness. The verification of the final signatures is justified by the following two equations:

$$\begin{aligned} e(\sigma, g) &= e((g_{k-1}^{\prod_{i \in [s]} a_{i,m_i} \cdot \prod_{i \in [s]} b_{i,m_{s+i}}})^\alpha, g) \\ &= g_k^{\alpha \cdot \prod_{i \in [s]} a_{i,m_i} \cdot \prod_{i \in [s]} b_{i,m_{s+i}}}. \end{aligned}$$

and

$$e(g^\alpha, A_{1,m_1}, \dots, A_{s,m_s}, B_{1,m_{s+1}}, \dots, B_{s,m_{k-1}}) = g_k^{\alpha \cdot \prod_{i \in [s]} a_{i,m_i} \cdot \prod_{i \in [s]} b_{i,m_{s+i}}}.$$

Theorem 5. *If the $k = (2s + 1)$ -MCDH assumption is hold in the multilinear groups, then the above PBS construction for left/right predicates and for messages of length s is selectively unforgeable and perfectly private.*

Selective Unforgeability. For length of messages $k - 1 = 2 \cdot s$, we prove security under the k -MCDH assumption. We show that if there exists a PPT adversary \mathcal{A} on our PBS scheme for messages of length $k - 1$ in the selective security game then we can construct an efficient algorithm \mathcal{B} on the k -MCDH assumption. We describe how \mathcal{B} interacts with \mathcal{A} .

The algorithm \mathcal{B} first receives a $k = (2 \cdot s + 1)$ -MCDH challenge consisting of the group sequence description $\vec{\mathbb{G}}$ and $g = g_1, g^{c_1}, \dots, g^{c_k}$. It also receives challenge message $m^* = m_1^* \dots m_s^* m_{s+1}^* \dots m_{k-1}^* \in \{0, 1\}^{k-1}$ from the adversary.

Setup: Initially, \mathcal{B} chooses random $u_1, \dots, u_s \in \mathbb{Z}_p$ and sets

$$A_{i,\beta} = \begin{cases} g^{c_i}, & \text{if } m_i^* = \beta \\ g^{u_i}, & \text{if } m_i^* \neq \beta \end{cases}$$

for $i \in [s], \beta \in \{0, 1\}$. This corresponds to setting $a_{i,\beta} = c_i$ if $m_i^* = \beta$ and u_i otherwise.

It also chooses random $v_1, \dots, v_s \in \mathbb{Z}_p$ and sets

$$B_{i,\beta} = \begin{cases} g^{c_{s+i}}, & \text{if } m_{s+i}^* = \beta \\ g^{v_i}, & \text{if } m_{s+i}^* \neq \beta \end{cases}$$

for $i \in [s], \beta \in \{0, 1\}$. This corresponds to setting $b_{i,\beta} = c_{s+i}$ if $m_{s+i}^* = \beta$ and v_i otherwise. We observe these are distributed identically to the real scheme. In addition, it will internally view $\alpha = c_k$.

Key Generation Oracle: The adversary \mathcal{A} will query for a secret key for a left predicate $p_x^{(L)}$ or a right predicate $p_y^{(R)}$.

If $p_x^{(L)}(m^*) = 1$ or $p_y^{(R)}(m^*) = 1$, then \mathcal{B} aborts the game.

Otherwise, $p_x^{(L)}(m^*) = p_y^{(R)}(m^*) = 0$, then \mathcal{B} will be able to create signing keys for the adversary, because his query will differ from the challenge message at least one bit. More specifically, for the left predicate $p_x^{(L)}$, we let $x_j \neq m_j^*, j \in [s]$, then \mathcal{B} produces the delegation key as $sk_{p_x^{(L)}} = e(g^{c_k}, g_{s-1}^{\prod_{i \neq j \in [s]} a_{i,x_i}}) u_j$; for the right predicate $p_y^{(R)}$, we let $y_j \neq m_{s+j}^*, j \in [s]$, then \mathcal{B} produces the signing key as $sk_{p_y^{(R)}} = e(g^{c_k}, g_{s-1}^{\prod_{i \neq j \in [s]} b_{i,y_i}}) v_j$.

Signing Oracle: The adversary \mathcal{A} will query for a signature for a message $m \neq m^*$, and we let $m_j \neq m_j^*, j \in [k - 1]$. Conceptually, \mathcal{B} will be able to create signature for the adversary, because his query will differ from the challenge message in at least one bit. More specifically, \mathcal{B} proceeds to make the signature according to the following two cases.

Case 1: If $j \in [s]$, \mathcal{B} produces the signature as:

$$\sigma = e(g^{c_k}, g_{k-2}^{\prod_{i \neq j \in [s]} a_{i,m_i} \cdot \prod_{i \in [s]} b_{i,m_{s+i}}}) u_j.$$

Case 2: If $j \in \{s + 1, \dots, k - 1\}$, \mathcal{B} produces the signature as:

$$\sigma = e(g^{c_k}, g_{k-2}^{\prod_{i \in [s]} a_{i,m_i} \cdot \prod_{i \neq j \in [s]} b_{i,m_{s+i}}}) v_j.$$

Forgery: Eventually, \mathcal{A} outputs a signature σ^* on message m^* . Then \mathcal{B} outputs σ^* as the solution of the given instance of the k -MCDH assumption.

According to the public parameters built in the setup phase and the assumption that σ^* is valid, we know that $\sigma^* = g_{k-1}^{\prod_{i \in [k]} c_i}$, implies that σ^* is a solution for the given instance of the k -MCDH problem, and thus \mathcal{B} breaks the k -MCDH assumption. It is clear that the view of \mathcal{A} simulated by \mathcal{B} in the above game is distributed statistically exponentially closely to that in the real unforgeability game, hence \mathcal{B} succeeds whenever \mathcal{A} does. \square

Perfect Privacy: Given a valid signature (m^*, σ^*) , we show that any signing key sk_p such that $p(m^*) = 1$ could possibly have created it. The proof is straightforward.

According to the setup of the signing algorithm, for any tuple (p_0, p_1, m^*) such that $p_0(m^*) = p_1(m^*) = 1$, which was chosen by any adversary \mathcal{A} , both of the signatures created by the signing key sk_{p_0} and sk_{p_1} are $(g_{k-1}^{\prod_{i \in [s]} a_{i,m_i^*} \cdot \prod_{i \in [s]} b_{i,m_{s+i}^*}})^\alpha$. Therefore, any signing key sk_p such that $p(m^*) = 1$ can compute a same signature on a given message m^* . The perfect privacy follows easily from this observation. \square

5.3 PBS for Circuit Predicates

We now construct a policy-based signature scheme for boolean circuit predicates. Our circuit notion is from [5], please refer to [5] for details.

Setup($1^\lambda, k = \ell + n + 1$): The setup algorithm takes as input a security parameter λ , the maximum depth ℓ of a circuit and the length of the message n (it also is the number of boolean inputs).

The algorithm then runs $\mathcal{G}(1^\lambda, k = n + \ell + 1)$ that produces groups $\vec{\mathbb{G}} = (\mathbb{G}_1, \dots, \mathbb{G}_k)$ of prime order p , with canonical generators g_1, \dots, g_k , where we let $g = g_1$. Next it chooses random $\alpha \in \mathbb{Z}_p$ and $(a_{1,0}, a_{1,1}), \dots, (a_{n,0}, a_{n,1}) \in \mathbb{Z}_p^2$ and computes $A_{i,\beta} = g^{\alpha_i, \beta}$ for $i \in [n], \beta \in \{0, 1\}$.

The master secret key is $msk = \alpha$. The public parameters, pk , consist of the group sequence description plus group elements $g_{\ell+1}^\alpha, \{A_{i,\beta} | i \in [n], \beta \in \{0, 1\}\}$ and message space $\mathcal{M} = \{0, 1\}^n$, signature space $\mathcal{S} = \mathbb{G}_{k-1}$, and predicate space $\mathcal{P} = \mathcal{C}$ that is the set of polynomial size circuit predicates.

KeyGen($sk, p = (n, q, A, B, \text{GateType})$): The key generation algorithm takes as input the master secret key msk and a description p of a circuit. The circuit has $n + q$ wires with n input wires, q gates and the $(n + q)$ -th wire designated as the output wire.

The key generation algorithm chooses random integers $r_1, \dots, r_{n+q-1} \in \mathbb{Z}_p$, where we think of the random value r_w as being associated with wire w . It sets $r_{n+q} = \alpha$.

Next, the algorithm generates key components for every wire w . The structure of the key components depends upon whether w is an input wire, an OR gate, or an AND gate. We describe how it generates components for each case.

- *Input wire.*

By our convention if $w \in [n]$ then it corresponds to the w -th input. The key component is:

$$K_w = g_2^{r_w a_{w,1}}.$$

- *OR gate.*

Suppose that wire $w \in \text{Gates}$ and that $\text{GateType}(w) = \text{OR}$. In addition, let $j = \text{depth}(w)$ be the depth of the wire. The algorithm will choose random $a_w, b_w \in \mathbb{Z}_p$. Then the algorithm creates key components as:

$$K_{w,1} = g^{a_w}, K_{w,2} = g^{b_w}, K_{w,3} = g_j^{r_w - a_w \cdot r_{A(w)}}, K_{w,4} = g_j^{r_w - b_w \cdot r_{B(w)}}.$$

- *AND gate.*

Suppose that wire $w \in \text{Gates}$ and that $\text{GateType}(w) = \text{AND}$. In addition, let $j = \text{depth}(w)$ be the depth of wire w . The algorithm chooses

random $a_w, b_w \in \mathbb{Z}_p$ and creates the key components as:

$$K_{w,1} = g^{a_w}, K_{w,2} = g^{b_w}, K_{w,3} = g_j^{r_w - a_w \cdot r_{A(w)} - b_w \cdot r_{B(w)}}.$$

The signing key sk_p consists of the description of p along with these $n + q$ key components.

Sign($sk_p, m \in \{0, 1\}^n$): The signing algorithm takes as input a signing key sk_p for a circuit predicate $p = (n, q, A, B, \text{GateType})$ and a message $m = m_1 \dots m_n$. The algorithm first checks that $p(m) = 1$; if not it aborts.

The goal of the algorithm is to compute the signature $\sigma = g_{n+\ell}^{\alpha \cdot \prod_{i \in [n]} a_{i, m_i}} \in \mathbb{G}_{n+\ell}$. We will compute the circuit from the bottom up.

- *Input wire.*

By our convention if $w \in [n]$ then it corresponds to the w -th input. Suppose that $m_w = p_w(m) = 1$. The algorithm computes $E_w = g_{n+1}^{r_w \cdot \prod_{i \neq w} a_{i, m_i}}$. Using the multilinear operation from A_{i, m_i} for $i \in [n] \neq w$. It then computes:

$$E_w = e(K_w, g_{n-1}^{\prod_{i \neq w} a_{i, m_i}}) = e(g_2^{r_w a_{w,1}}, g_{n-1}^{\prod_{i \neq w} a_{i, m_i}}) = g_{n+1}^{r_w \prod_{i \in [n]} a_{i, m_i}}.$$

- *OR gate.*

Consider a wire $w \in \text{Gates}$ and that $\text{GateType}(w) = \text{OR}$. In addition, let $j = \text{depth}(w)$ be the depth of the wire. For exposition we define $D(m) = g_n^{\prod_{i \in [n]} a_{i, m_i}}$. This is computable via the multilinear operation from A_{i, m_i} for $i \in [n]$. The computation is performed if $p_w(m) = 1$. If $p_{A(w)}(m) = 1$ (i.e., the first input evaluated to 1) then it computes:

$$\begin{aligned} E_w &= e(E_{A(w)}, K_{w,1}) \cdot e(K_{w,3}, D(m)) \\ &= e(g_{j+n-1}^{r_{A(w)} \prod_{i \in [n]} a_{i, m_i}}, g^{a_w}) \\ &\quad \cdot e(g_j^{r_w - a_w \cdot r_{A(w)}}, g_n^{\prod_{i \in [n]} a_{i, m_i}}) \\ &= g_{j+n}^{r_w \prod_{i \in [n]} a_{i, m_i}}. \end{aligned}$$

Otherwise, if $p_{A(w)}(m) = 0$ but $p_{B(w)}(m) = 1$, then it computes:

$$\begin{aligned} E_w &= e(E_{B(w)}, K_{w,2}) \cdot e(K_{w,4}, D(m)) \\ &= e(g_{j+n-1}^{r_{B(w)} \prod_{i \in [n]} a_{i, m_i}}, g^{b_w}) \\ &\quad \cdot e(g_j^{r_w - b_w \cdot r_{B(w)}}, g_n^{\prod_{i \in [n]} a_{i, m_i}}) \\ &= g_{j+n}^{r_w \prod_{i \in [n]} a_{i, m_i}}. \end{aligned}$$

- *AND gate.*

Consider a wire $w \in \text{Gates}$ and that $\text{GateType}(w) = \text{AND}$. In addition, let $j = \text{depth}(w)$ be the depth of the wire. The computation is performed if $p_w(m) = 1$ (i.e.,

$p_{A(w)}(m) = p_{B(w)}(m) = 1$) then it computes:

$$\begin{aligned} E_w &= e(E_{A(w)}, K_{w,1}) \cdot e(E_{B(w)}, K_{w,2}) \\ &\quad \cdot e(K_{w,3}, D(m)) \\ &= e(g_{j+n-1}^{r_{A(w)} \prod_i a_i, m_i}, g^{a_w}) \cdot e(g_{j+n-1}^{r_{B(w)} \prod_i a_i, m_i}, g^{b_w}) \\ &\quad \cdot e(g_j^{r_w - a_w \cdot r_{A(w)} - b_w \cdot r_{B(w)}}, g_n^{\prod_i a_i, m_i}) \\ &= g_{j+n}^{r_w \prod_i a_i, m_i}. \end{aligned}$$

The above procedures are evaluated in order for all w for which $p_w(m) = 1$. The final output of these procedures gives a signature:

$$\sigma = g_{n+\ell}^{r_{n+q} \prod_{i \in [n]} a_i, m_i} = g_{n+\ell}^{\alpha \cdot \prod_{i \in [n]} a_i, m_i} \in \mathbb{G}_{n+\ell}.$$

Verify($pk, m \in \{0, 1\}^n, \sigma \in \mathbb{G}_{k-1}$): Given a purported signature σ on a message m , verify the following equation:

$$e(\sigma, g) = e(g_{\ell+1}^\alpha, A_{1, m_1}, \dots, A_{n, m_n}).$$

Output 1 if it holds, else 0.

Correctness. The verification of the signature is justified by the following two equations:

$$\begin{aligned} e(\sigma, g) &= e(g_{n+\ell}^{\alpha \cdot \prod_{i \in [n]} a_i, m_i}, g) \\ &= g_{n+\ell+1}^{\alpha \cdot \prod_{i \in [n]} a_i, m_i}. \end{aligned}$$

and

$$e(g_{\ell+1}^\alpha, A_{1, m_1}, \dots, A_{n, m_n}) = g_{n+\ell+1}^{\alpha \cdot \prod_{i \in [n]} a_i, m_i}.$$

Theorem 6. *If the $k = (n + \ell + 1)$ -MCDH assumption holds in the multilinear groups, then the above PBS construction for arbitrary circuits of depth ℓ and input length n , and messages of length n is selectively unforgeable and perfectly private.*

Selective Unforgeability. For length of messages n and a circuit of max depth ℓ and input length n , we prove security under the $k = (n + \ell + 1)$ -Multilinear Computational Diffie-Hellman assumption.

We show that if there exists a PPT adversary \mathcal{A} on our PBS scheme for messages of length s and circuits of depth ℓ and inputs of length n in the selective security game then we can construct an efficient algorithm \mathcal{B} on the $(n + \ell + 1)$ -MCDH assumption. We describe how \mathcal{B} interacts with \mathcal{A} .

The algorithm \mathcal{B} first receives a $k = (n + \ell + 1)$ -MCDH challenge consisting of the group sequence description $\vec{\mathbb{G}}$ and $g = g_1, g^{c_1}, \dots, g^{c_k}$. It also receives challenge attribute message $m^* \in \{0, 1\}^n$ from the adversary \mathcal{A} .

Setup: Initially, \mathcal{B} chooses random $u_1, \dots, u_n \in \mathbb{Z}_p$ and sets

$$A_{i, \beta} = \begin{cases} g^{c_i}, & \text{if } m_i^* = \beta \\ g^{u_i}, & \text{if } m_i^* \neq \beta \end{cases}$$

for $i \in [n], \beta \in \{0, 1\}$. This corresponds to setting $a_{i, \beta} = c_i$ if $m_i^* = \beta$ and u_i otherwise. In addition, it will internally view $\alpha = c_{n+1} \cdot c_{n+2} \cdots c_{n+\ell+1}$.

Key Generation Oracle: The adversary \mathcal{A} will query for a signing key for a circuit $p = (n, q, A, B, \text{GateType})$, where $p(m^*) = 0$. \mathcal{B} proceeds to make the key. The idea for this oracle is same as in [5]. We will think have some invariant properties for each gate. Consider a gate w at depth j and the simulators viewpoint (symbolically) of r_w . If $p_w(x^*) = 0$, then the simulator will view r_w as the term $c_{n+1} \cdot c_{n+2} \cdots c_{n+j+1}$ plus some additional known randomization terms. If $p_w(x^*) = 1$, then the simulator will view r_w as the 0 plus some additional known randomization terms. If we can keep this property intact for simulating the keys up the circuit, the simulator will view r_{n+q} as $c_{n+1} \cdot c_{n+2} \cdots c_{n+\ell}$.

We describe how to create the key components for each wire w . Again, we organize key component creation into input wires, OR gates, and AND gates.

- *Input wire.*

Suppose $w \in [n]$ and is therefore by convention an input wire.

- * If $(m^*)_w = 1$ then we choose random $r_w \leftarrow \mathbb{Z}_p$ (as is done honestly). The key component is:

$$K_w = g_2^{r_w a_w, 1}.$$

- * If $(m^*)_w = 0$ then we let $r_w = c_{n+1} c_{n+2} + \eta_w$ where $\eta_w \in \mathbb{Z}_p$ is a randomly chosen value. The key component is:

$$K_w = (e(g^{c_{n+1}}, g^{c_{n+2}}) \cdot g_2^{\eta_w})^{a_w} = g_2^{r_w a_w, 1}.$$

- *OR gate.*

Suppose that wire $w \in \text{Gates}$ and that $\text{GateType}(w) = \text{OR}$. In addition, let $j = \text{depth}(w)$ be the depth of the wire.

- * If $p_w(x^*) = 1$, then algorithm will choose random $a_w, b_w, r_w \in \mathbb{Z}_p$. Then the algorithm creates key components as:

$$\begin{aligned} K_{w,1} &= g^{a_w}, K_{w,2} = g^{b_w}, K_{w,3} = \\ &g_j^{r_w - a_w \cdot r_{A(w)}}, K_{w,4} = g_j^{r_w - b_w \cdot r_{B(w)}}. \end{aligned}$$

- * If $p_w(x^*) = 0$, then we set $a_w = c_{n+j+1} + \psi_w, b_w = c_{n+j+1} + \phi_w$, and $r_w = c_{n+1} \cdot c_{n+2} \cdots c_{n+j+1} + \eta_w$, where ψ_w, ϕ_w, η_w are chosen randomly. Then the algorithm creates key components as:

$$\begin{aligned} K_{w,1} &= g^{c_{n+j+1} + \psi_w}, K_{w,2} = g^{c_{n+j+1} + \phi_w}, \\ &K_{w,3} = \\ &g_j^{\eta_w - c_{n+j+1} \eta_{A(w)} - \psi_w (c_{n+1} \cdots c_{n+j} + \eta_{A(w)})}, \\ &K_{w,4} = \\ &g_j^{\eta_w - c_{n+j+1} \eta_{B(w)} - \phi_w (c_{n+1} \cdots c_{n+j} + \eta_{B(w)})}. \end{aligned}$$

\mathcal{B} can create the last two key components due to a cancellation. Since both the $A(w)$ and $B(w)$ gates evaluated to 0, we have $r_{A(w)} = c_{n+1} \cdots c_{n+j} + \eta_{A(w)}$ and similarly for $r_{B(w)}$. Note that $g_j^{c_{n+1} \cdots c_{n+j}}$ is always using the multilinear maps.

- **AND gate.**

Suppose that wire $w \in \text{Gates}$ and that $\text{GateType}(w) = \text{AND}$. In addition, let $j = \text{depth}(w)$ be the depth of wire w .

- * If $p_w(x^*) = 1$, then the algorithm chooses random $a_w, b_w, r_w \in \mathbb{Z}_p$ and creates the key components as:

$$K_{w,1} = g^{a_w}, K_{w,2} = g^{b_w}, \\ K_{w,3} = g_j^{r_w - a_w \cdot r_{A(w)} - b_w \cdot r_{B(w)}}.$$

- * If $p_w(x^*) = 0$ and $p_{A(w)}(x^*) = 0$, then we let $a_w = c_{n+j+1} + \psi_w, b_w = \phi_w$, and $r_w = c_{n+1} \cdot c_{n+2} \cdots c_{n+j+1} + \eta_w$, where ψ_w, ϕ_w, η_w are chosen randomly. Then the algorithm creates key components as:

$$K_{w,1} = g^{c_{n+j+1} + \psi_w}, K_{w,2} = g^{\phi_w}, \\ K_{w,3} = g_j^{\eta_w - \psi_w c_{n+1} \cdots c_{n+j} - (c_{n+j+1} + \psi_w) \eta_{A(w)} - \phi_w r_{B(w)}}.$$

\mathcal{B} can create the last key component due to a cancellation. Since the $A(w)$ gate evaluated to 0, we have $r_{A(w)} = c_{n+1} \cdots c_{n+j} + \eta_{A(w)}$. Note that $g_j^{r_{B(w)}}$ always computable regardless of whether $p_{A(w)}(x^*)$ evaluated to 0 or 1, since $g_j^{c_{n+1} \cdots c_{n+j}}$ is always using the multilinear maps.

The case where $p_{B(w)}(x^*) = 0$ and $p_{A(w)}(x^*) = 1$ is performed in a symmetric to what is above, with the roles of a_w and b_w reversed.

Signing Oracle: The adversary \mathcal{A} will query for a signature for a message $m \neq m^*$, and we let $m_j \neq m_j^*$.

\mathcal{B} can produce a valid signature $\sigma = (g_{n+\ell}^{\prod_{i \neq j \in [k]} c_i})^{u_j}$ by knowing the exponent u_j .

Forgery: Eventually, \mathcal{A} outputs an attribute signature σ^* on message m^* . Then \mathcal{B} outputs σ^* as the solution of the given instance of the $k = (n+\ell+1)$ -MCDH assumption. According to the public key built in the setup phase and the assumption that σ^* is valid, we know that $\sigma^* = g_{k-1}^{\prod_{i \in [k]} c_i}$, implies that σ^* is a solution for the given instance of the k -MCDH problem, and thus \mathcal{B} breaks the k -MCDH assumption.

It is clear that the view of \mathcal{A} simulated by \mathcal{B} in the above game is distributed statistically exponentially closely to that in the real unforgeability game, hence \mathcal{B} succeeds whenever \mathcal{A} does. \square

Perfect Privacy. Given a valid signature ($m^* \in \{0, 1\}^n, \sigma^* \in \mathbb{G}_{k-1}$), we show that any signing key sk_p such that $p(m^*) = 1$ could possibly have created it. The proof is straight-forward.

According to the setup of the signing algorithm, for any tuple (p_0, p_1, m^*) such that $p_0(m^*) = p_1(m^*) = 1$, which was chosen by an unbounded adversary \mathcal{A} , both of the signatures created by the signing key sk_{p_0} and sk_{p_1} are $g_{n+\ell}^{\alpha \cdot \prod_{i \in [n]} a_i \cdot m_i^*}$. Therefore, any signing key sk_p such

that $p(m^*) = 1$ can compute a same signature on a given message m^* . The perfect privacy follows easily from this observation. \square

6 Conclusion

In this work, we introduce the notion of policy-based signature for predicates. In such a signature scheme, signers can sign messages that conform to some predicate, yet privacy of the predicate is maintained. Then, we construct a policy-based signature scheme for prefix predicate based on tree-based signature scheme. Furthermore, we also construct several policy-based signature schemes for bit-fixing predicate, left/right predicate, and circuits predicate, respectively, based on multilinear maps.

Acknowledgments

This study was supported by the Science and technology research project of Chongqing Municipal Education Commission (No. KJ1600445). The authors gratefully acknowledge the anonymous reviewers for their valuable comments.

References

- [1] M. Bellare and G. Fuchsbauer, "Policy-based signatures", *The 17th IACR International Conference on Practice and Theory of Public-Key Cryptography (PKC'14)*, pp. 520–537, Buenos Aires, Argentina, 2014.
- [2] E. Boyle, S. Goldwasser, and I. Ivan, "Functional signatures and pseudorandom functions", *The 17th IACR International Conference on Practice and Theory of Public-Key Cryptography (PKC'14)*, pp. 501–519, Buenos Aires, Argentina, 2014.
- [3] X. Boyen, "Mesh signatures", *The 26th Annual International Conference on the Theory and Applications of Cryptographic Techniques (EUROCRYPT'07)*, pp. 210–227, Barcelona, Spain, 2007.
- [4] D. Boneh and A. Silverberg, "Applications of multilinear forms to cryptography", *Contemporary Mathematics*, vol. 324, no. 1, pp. 71–90, 2003.
- [5] D. Boneh and B. Waters, "Constrained pseudorandom functions and their applications", *The 19th International Conference on the Theory and Application of Cryptology and Information Security (ASIACRYPT'13)*, pp. 280–300, Bengaluru, India, 2013.
- [6] D. Chaum and E. Van Heyst, "Group signatures", *Workshop on the Theory and Application of Cryptographic Techniques (EUROCRYPT'91)*, pp. 257–265, Brighton, UK, 1991.
- [7] M. L. Chande, C. C. Lee, and C. T. Li, "Message recovery via an efficient multi-proxy signature with self-certified keys", *International Journal of Network Security*, vol. 19, no. 3, pp. 340–346, 2017.

- [8] J. S. Coron, T. Lepoint, and M. Tibouchi, “Practical multilinear maps over the integers”, *The 33rd Annual Cryptology Conference (CRYPTO’13)*, pp. 476–493, Santa Barbara, CA, USA, 2013.
- [9] S. Garg, C. Gentry, and S. Halevi, “Candidate multilinear maps from ideal lattices”, *The 32nd Annual International Conference on the Theory and Applications of Cryptographic Techniques (EUROCRYPT’13)*, pp. 1–17, Athens, Greece, 2013.
- [10] C. Gentry, S. Gorbunov, and S. Halevi, “Graph-induced multilinear maps from lattices”, *The 12th IACR Theory of Cryptography Conference (TCC’15)*, pp. 498–527, Warsaw, Poland, 2015.
- [11] J. Groth and A. Sahai, “Efficient non-interactive proof systems for bilinear groups”, *The 27th Annual International Conference on the Theory and Applications of Cryptographic Techniques (EUROCRYPT’08)*, pp. 415–432, Istanbul, Turkey, 2008.
- [12] Y. Hu and H. Jia, “Cryptanalysis of GGH map”, *The 35th Annual International Conference on the Theory and Applications of Cryptographic Techniques (EUROCRYPT’16)*, pp. 537–565, Vienna, Austria, 2016.
- [13] S. Hohenberger, A. Sahai, and B. Waters, “Full domain hash from (leveled) multilinear maps and identity-based aggregate signatures”, *The 33rd Annual Cryptology Conference (CRYPTO’13)*, pp. 494–512, Santa Barbara, CA, USA, 2013.
- [14] M. Ibrahim, “Resisting traitors in linkable democratic group signatures”, *International Journal of Network Security*, vol. 9, no. 1, pp. 51–60, 2009.
- [15] E. Kiltz, A. Mityagin, S. Panjwani, and B. Raghavan, “Append-only signatures”, *The 32nd International Colloquium on Automata, Languages and Programming (ICALP’05)*, pp. 434–445, Lisbon, Portugal, 2005.
- [16] L. Lamport, “Constructing digital signatures from a one-way function”, *Technical Report SRI-CSL-98*, SRI Intl. Computer Science Laboratory, Oct. 1979.
- [17] C. C. Lee, T. C. Lin, S. F. Tzeng, M. S. Hwang, “Generalization of proxy signature based on factorization”, *International Journal of Innovative Computing, Information and Control*, vol. 7, no. 3, pp. 1039–1054, 2011.
- [18] H. T. Lee and J. H. Seo, “Security analysis of multilinear maps over the integers”, *The 34th Annual Cryptology Conference (CRYPTO’14)*, pp. 224–240, Santa Barbara, CA, USA, 2014.
- [19] R. C. Merkle, “A certified digital signature (that antique paper from 1979)”, *Annual Cryptology Conference (CRYPTO’89)*, pp. 218–238, Santa Barbara, CA, USA, 1989.
- [20] H. K. Maji, M. Prabhakaran, and M. Rosulek, “Attribute-based signatures”, *The Cryptographers’ Track at the RSA Conference (CT-RSA’11)*, pp. 376–392, San Francisco, CA, USA, 2011.
- [21] M. Mambo, K. Usuda, and E. Okamoto, “Proxy signatures for delegating signing operation”, *The 3rd ACM conference on computer and communications security (CCS’96)*, pp. 48–57, New Delhi, India, 1996.
- [22] Z. Qin, H. Xiong, and F. Li, “A Provably Secure Certificate Based Ring Signature Without Pairing”, *International Journal of Network Security*, vol. 16, no. 4, pp. 278–285, 2014.
- [23] R. L. Rivest, A. Shamir, and Y. Tauman, “How to leak a secret”, *The 7th International Conference on the Theory and Application of Cryptology and Information Security (ASIACRYPT’01)*, pp. 552–565, Gold Coast, Australia, 2001.
- [24] F. Tang, H. Li, and B. Liang, “Attribute-based signature for circuits from multilinear maps”, *The 17th Information Security Conference (ISC’14)*, pp. 54–71, Hongkong, China, 2014.
- [25] 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

Fei Tang received his Ph.D. from the Institute of Information Engineering of Chinese Academy of Sciences in 2015. He is currently a lecturer of the College of Cyberspace Security and Law, Chongqing University of Posts and Telecommunications. His research interest is public key cryptography.

Yousheng Zhou is currently an associate professor of the College of Cyberspace Security and Law, Chongqing University of Posts and Telecommunications. He received his Ph.D. from Beijing University of Posts and Telecommunications in 2011. He completed one-year postdoctorate work at Dublin City University, Ireland in 2016. His research interests include network security and cloud security.

A New Way to Prevent UKS Attacks Using Hardware Security Chips

Qianying Zhang^{1,2}, Zhiping Shi^{1,3}

(Corresponding author: Qianying Zhang)

College of Information Engineering¹

Beijing Advanced Innovation Center for Imaging Technology²

Beijing Key Laboratory of Electronic System Reliability Technology³

Capital Normal University, Beijing 100048, China

(Email: qyzhang@cnu.edu.cn)

(Received Nov. 30, 2016; revised and accepted Feb. 13 & Mar. 11, 2017)

Abstract

UKS (unknown key-share) attacks are common attacks on AKE (Authenticated Key Exchange) protocols. We summarize two common countermeasures against UKS attacks on a kind of AKE protocols whose message flows are basic Diffie-Hellman exchanges. The first countermeasure forces the CA to check the possession of private key during registration, which is impractical for the CA. The second countermeasure adds identities in the derivation of the session key, which leads to modification of the protocols which might already be standardized and widely used in practice. By using protection of cryptographic keys provided by hardware security chips, such as TPM or TCM, we propose a new way that requires no check of possession of private key and no addition of identity during the derivation of the session key to prevent UKS attacks. We modify the CK model to adapt protocols using hardware security chip. We then implement a protocol once used in NSA, called KEA and subject to UKS attacks, using TCM chips. Our implementation, called tKEA, without forcing the CA to check during registration and modifying the original KEA, is proven to be secure. To show the generality of our way, we also show that it can prevent UKS attacks on the MQV protocol.

Keywords: Authenticated Key Exchange; CK Model; KEA; Trusted Cryptography Module; UKS Attacks

1 Introduction

The key exchange protocol, first proposed by Diffie and Hellman [11], allows two entities to establish a shared secret key via public communication. In order to authenticate identities of the two entities involved in the protocol, authenticated key exchange (AKE) is proposed. AKE not only allows two entities to compute a shared secret key but also ensures the authenticity of entities.

To date, a great number of AKE protocols have been proposed [1, 2, 3, 8, 12, 13, 14, 15, 19, 25, 27, 31] and many of them are subsequently broken, such as KEA [25] and MQV [22, 23]. KEA was designed by NSA (National Security Agency) in 1994 and kept secret until 1998. Microsoft researchers K.Lauter and A.Mityagin find that the original KEA protocol is susceptible to UKS attacks [21]. Then they present a modified version of KEA protocol, called KEA+ [21], which is resistant to UKS attacks, and give a formal proof. The MQV protocol is a famous and efficient AKE protocol designed by Law, Menezes, Qu, Solinas and Vanstone. This protocol was found to be susceptible to UKS attacks by Kaliski [17], then Krawczyk found that MQV hold none of its stated security goals, such as resistance to KCI attacks and the security property of perfect forward secrecy (PFS). To achieve the security goals of MQV, Krawczyk proposes a hashed variant of MQV, HMQV [19].

1.1 Related Work

In order to formally prove that AKE protocols are secure, Bellare and Rogaway in 1993 provided the first formal definition for an AKE model [4], which we refer to as the BR model. After that, a lot of variants of BR model were represented and many authenticated key exchange protocols were proposed. For more details, we refer the readers to [10] for a comparison and discussion of variants models for authenticated key exchange. Based on BR model, Canetti and Krawczyk proposed the CK model [7], based on which the HMQV protocol was proved. LaMacchia, Lauter, and Mityagin defined a new model called eCK [20], which is much stronger than BR and CK models. They also introduced a new AKE protocol called NAXOS and proved its security in eCK model. However the NAXOS protocol is less efficient in that it requires 4 exponentiations per entity compared to 3 exponentiations for KEA.

Katz [18] first gives the idea of using secure hardware to achieve stronger security properties, and proves that tamper-proof hardware suffices to circumvent the impossibility result of secure computation of general functionalities without an honest majority. Recently, some works extend this idea to the improvement and security analysis of AKE protocols on modern AKE security models. [30] analyzes the SM2 key exchange protocol in TPM 2.0 security chip [29], and shows that protection provided by the TPM security chip indeed helps the protocol to resist two kinds of UKS attacks. [33] leverages the tamper-proof hardware to protect cryptographic keys and designs a set of APIs for HMQV, and formally proves that the HMQV protocol achieves full PFS property with the help of tamper-proof hardware in the CK model. [31] proposes an efficient key exchange protocol called sHMQV, which is a variant of HMQV. sHMQV eliminates the validation of public ephemeral key by protecting the ephemeral private key in trusted hardware devices, and enjoys the best efficiency in current one-round key exchange protocols. [32] models the protection provided by TPM 2.0 security chip as an oracle, and formally proves that under the protection of TPM 2.0 the key exchange primitive in TPM 2.0 is secure in modern AKE model.

1.2 UKS Attacks

A UKS attack on an AKE protocol is that an entity A ends up believing that he shares a key with an entity B, and although this is in fact the case, B mistakenly believes that the key is shared with an entity $E \neq A$. Since the adversary E does not obtain the shared secret key, he cannot modify or decrypt the messages between A and B. However, E can take advantage of the entities' false assumptions about the identity who shares the key. Take a scenario described in [12] for example: B is a bank, and A sends him a digital coin, encrypted with the shared secret key, for deposit into her account. Believing that the key is shared with E, B assumes the coin is from E and deposits it into E's account instead. Several UKS attacks have been proposed in the literature, such as attacks on STS [5], KEA [21], and MQV [17].

1.3 Contributions

We give our contributions as follows:

- 1) We summarize UKS attacks on AKE protocols and existing countermeasures in the literature, and identify two kinds of attacks, called *public key substitution UKS attack* and *public key registration UKS attack* respectively. The details of the two attack are described in Section 2. The usual way to resist the two kinds of UKS attacks are: 1) force the CA to check the possession of the private key, 2) add the identity during the derivation of the session key. We illustrate these countermeasures by overview of existing works on preventing UKS attacks on KEA and MQV.

- 2) We present a new way to prevent the two kinds of UKS attacks using the protection capability of hardware security chip, such as Trusted Platform Module (TPM) [29] and Trusted Cryptography Module (TCM) [26]. The key idea is to make use of the security chip to generate the long-term secret key, and register it to a CA who does not check the possession of the private key and only makes sure that the key comes from a real hardware security chip. The protection capability of the security chip prevents the adversary from getting the plaintext of the private key even he corrupts and controls the security chip. In our security analysis we will show that the protection capability is crucial for the KEA protocol to resist UKS attacks. Our new way of preventing UKS attacks has advantages of not requiring the CA to check the possession of the private key nor modifying the original protocol. The former advantage makes the protocol can be deployed in practical CAs who usually do not check the possession of the private key. The later advantage improves the security of such protocols that have already been standardized and deployed in many fields. Upgrading the standards might require quite a long time, and in some fields system upgrades are rigorously controlled, such as the industrial control field. To show the generality of our way to resist UKS attacks, we also demonstrate that our proposed way can prevent UKS attacks on MQV protocol.

- 3) We give a variant of CK model to adapt protocols implemented by hardware security chip. Then we implement the KEA protocol (subject to UKS attacks) using TCM chips and prove that our implementation prevents UKS attacks. We make a comparison among typical protocols with the ability of resisting UKS attacks in terms of key registration (whether adversary-controlled entities can register arbitrary public keys), modification (whether the original protocol is modified in order to be proven secure formally), efficiency (whether extra computation is added), security properties and assumptions in Table 1. The Modif column shows that both KEA+ and HMQV modify the original protocols (KEA and MQV respectively) in order to be proven secure formally. The Effic column shows that the HMQV adds 25% extra computation to MQV while tKEA adds no extra computation to KEA. Compared to KEA+ and HMQV, tKEA obtains same security properties while making no modification of the original protocol and adding no extra computation.

1.4 Organization

We summarize the two kinds of UKS attacks and corresponding countermeasures in Section 2. In Section 3, we give a detailed description of protection of cryptographic keys provided by one kind of hardware security

Table 1: Comparison of HMQV, KEA+ and tKEA

	Key Reg.	Modif.	Effic.	Security	Assumptions
tKEA	Arbitrary	No	No	CK, KCI, wPFS	GDP+RO
KEA+	Arbitrary	Yes	No	CK, KCI, wPFS	GDP+RO
HMQV	Arbitrary	Yes	25%	CK, KCI, wPFS	GDP+KEA1+RO

chip, TCM, show that how it can be used on AKE protocols, and give our implementation, which we call tKEA (the ‘t’ stands for trusted). Section 4 describes the security model on which the formal security analysis of tKEA is based. Section 5 proves the security of tKEA. We also show how the protection provided by TCM prevents the UKS attack on MQV protocol described in [19] in this section. We end the paper with concluding and our future work in Section 6.

2 UKS Attacks and Their Countermeasures

AKE protocols can be categorized as the explicitly authenticated or the implicitly authenticated by the way they are authenticated. Both of the two kinds of AKE protocols are vulnerable to UKS attacks. Baek and Kim have summarized UKS attacks on the explicitly authenticated key exchange protocol [21]. In this paper we give an overview of UKS attacks on the implicitly authenticated key exchange protocol.

In this section, we first introduce the explicitly and implicitly authenticated key exchange protocols, and then summarize two kinds of UKS attacks on the implicitly authenticated key exchange protocol and the usual countermeasures to prevent these two kinds of attacks.

2.1 Explicitly Authenticated Key Exchange Protocol

The explicitly authenticated key exchange protocol is such a kind of protocol that first executes a basic Diffie-Hellman key exchange and then uses digital signatures or additional authenticating message flows to provide authentication explicitly. ISO-DH [16], STS [12], SIG-DH [28], SIGMA [8] are such typical protocols. In the following, we take the ISO-DH protocol as an example to illustrate such kind of protocol.

Let G be a group of primer order and denote by g a generator of G . Assume that entities have secret/public keys for some digital signature scheme SIG and that entities know each other’s registered public keys. The hat notation, such as \hat{A} , denotes the identities of entities in the protocol. Denote the signature of a message \mathcal{M} under the secrete key of an entity \hat{A} by $SIG_{\hat{A}}(\mathcal{M})$. We depict the protocol in Figure 1. First, an entity \hat{A} as an initiator randomly generates an ephemeral private key x and sends a tuple $\{g^x, SIG_{\hat{A}}(g^x, \hat{B})\}$ to \hat{B} , the responder. The responder \hat{B} generates an ephemeral private key

y and replies with a tuple $\{g^y, SIG_{\hat{B}}(g^y, g^x \hat{A})\}$. Both \hat{A} and \hat{B} then verify each other’s signatures, and compute a shared session key $K = g^{xy}$ if the verification successes.

2.2 Implicitly Authenticated Key Exchange Protocol

The implicitly authenticated key exchange protocol only needs basic Diffie-Hellman exchanges, and provides authentication by combining ephemeral keys and long-term keys during the derivation of the session key. KEA and MQV are typical protocols of this kind of AKE. Figure 2 gives an illustration of KEA and its variant KEA+. KEA involves two entities, \hat{A} and \hat{B} , with respective secret keys a and b and public keys g^a and g^b . KEA assumes that entities know each other’s registered public keys. The protocol first runs a Diffie-Hellman key exchange: \hat{A} and \hat{B} each generates its ephemeral private key, x and y respectively, and exchanges the ephemeral public keys g^x and g^y . Then each entity computes g^{ay} and g^{bx} and computes a session key by applying a hash function H to (g^{ay}, g^{bx}) . The KEA+ protocol differs from KEA when computing the session key, it applies the hash function to a tuple $(g^{ay}, g^{bx}, \hat{A}, \hat{B})$, adding the identities to the tuple of KEA.

2.3 UKS Attacks on Implicitly Authenticated Key Exchange Protocol

As Baek and Kim have given a conclusion of UKS attacks on the explicitly authenticated key exchange protocol [21], here we only summarize UKS attacks on the implicitly key exchange protocol. We categorize these attacks as *public key substitution UKS attack* and *public key registration UKS attack*. We also summarize existing countermeasures on the two kinds of UKS attacks.

2.3.1 Public Key Substitution UKS Attack

This kind of attack happens to some protocols when the CA does not check the possession of the private key. In the following we illustrate this attack on the KEA protocol. Consider two entities \hat{A} and \hat{B} preparing to start a session. An adversary \mathcal{M} registers a public key g^a of \hat{A} as his own public key. Then \mathcal{M} intercepts the session between \hat{A} and \hat{B} . \mathcal{M} forwards the ephemeral public key g^x from \hat{A} to \hat{B} and ephemeral public key g^y from \hat{B} to \hat{A} . Since \mathcal{M} has the same public key as \hat{A} , both \hat{A} and \hat{B} will compute identical session keys. However, \hat{A} completes a session with \hat{B} and \hat{B} completes a session with \mathcal{M} .

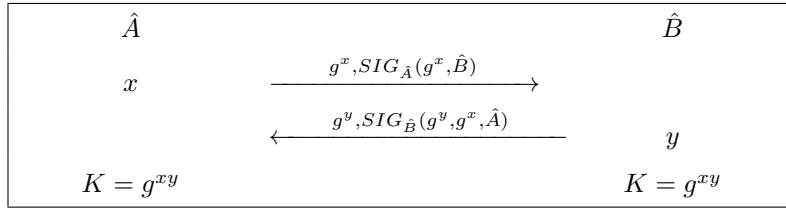


Figure 1: Explicitly authenticated key exchange protocol: ISO-DH

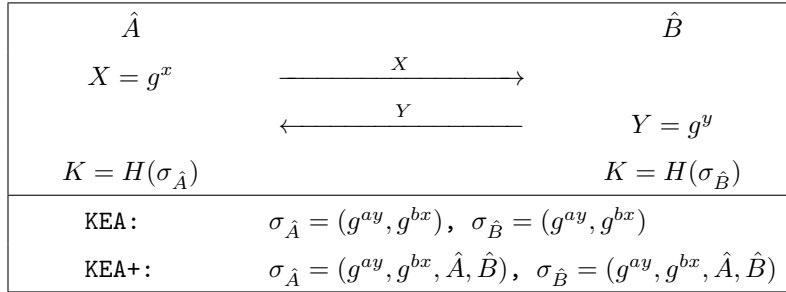


Figure 2: Implicitly authenticated key exchange protocol: KEA and KEA+

The usual way to solve this kind of UKS attack is to force the CA to check the possession of the private key. If the CA does, \mathcal{M} cannot register the public key of \hat{A} , then \hat{A} and \hat{B} will compute non-identical session keys. However, as the proof of knowledge check are rarely done by CA in practice, this way to prevent UKS attacks are impractical.

2.3.2 Public Key Registration UKS Attack

The typical attack example is a UKS attack on MQV found by Kaliski [17]. Let me introduce MQV first. MQV is a famous implicitly authenticated key exchange protocol, which was stated to have a lot of security properties, such as resistance to UKS attacks and KCI attacks. We depict MQV and HMQV in Figure 3. Entities \hat{A} and \hat{B} have their private/public key pairs (a, g^a) and (b, g^b) respectively. The ephemeral public keys in their exchange messages are g^x and g^y . The computation of the session key by \hat{A} (\hat{B}) is a hash value to $(YB^e)^{x+da} ((XA^d)^{y+be})$. The only difference between MQV and HMQV is the computation of d and e . The former only uses the ephemeral public key, while the later adds the identity information and uses a hash function in the computation. However, we will show below that this slight modification is crucial for the security of HMQV.

We describe the *public key registration UKS attack* on MQV in Figure 4. An adversary \mathcal{M} intercepts the ephemeral public key $X = g^x$ sent from \hat{A} to \hat{B} . Based on X , \mathcal{M} computes a private/public key pair (c, g^c) , and sends an ephemeral public key Z . After receiving Z , \hat{B} generates a random ephemeral key $Y = g^y$ and sends it to \mathcal{M} . \mathcal{M} transmits Y to \hat{A} . We denote the session between \hat{A} to \hat{B} by s , and the session between \hat{B} to \mathcal{M} by s' . We can see that the key pair (c, g^c) and the ephemeral key Z are computed so cleverly that s and s' have the identical shared secret key.

From the attack described above, we can see that check proof of knowledge of private key cannot prevent this attack as the the adversary holds the private key c that he registers. The usual way to prevent this kind of attack is to add the identities in the derivation of the session key. Krawczyk and Menezes respectively present HMQV and a variant of MQV [24] which both resist this kind of UKS attack. HMQV adds the identity and uses a hash function when computing d and e , while [24] adds the identities in the derivation of the session key. From their solutions we can see that adding identities in the derivation of the session key is an effective way to prevent the *public key registration UKS attack*. Although it might be easy to modify the protocol to achieve a higher security level, for protocols that have been standardized it might take a long time for them to be upgraded. So we need to consider how to protect systems adopting non-secure protocols while upgrades of protocols are still unavailable. And for some fields, such as industrial control field, upgrades are rigorously controlled as the system deals with very crucial tasks involving electricity and other infrastructures and any modification must be tested rigorously. So research on improving the security of AKE protocols without modifying the original protocol is meaningful.

3 Protection Provided by TCM

Trusted Cryptography Module (TCM), a hardware security chip similar to Trusted Platform Module (TPM), is a small tamper-resistant cryptographic chip embedded in computer platforms (e.g. on a PC motherboard). TCM provides a set of cryptography capabilities that allow some cryptographic functions to be executed in TCM, such as public-key decryption/encryption (SM2-1), hash (SM3), random number generating, key exchange protocol (SM2-2) and so on. TCM stores the secret data, such

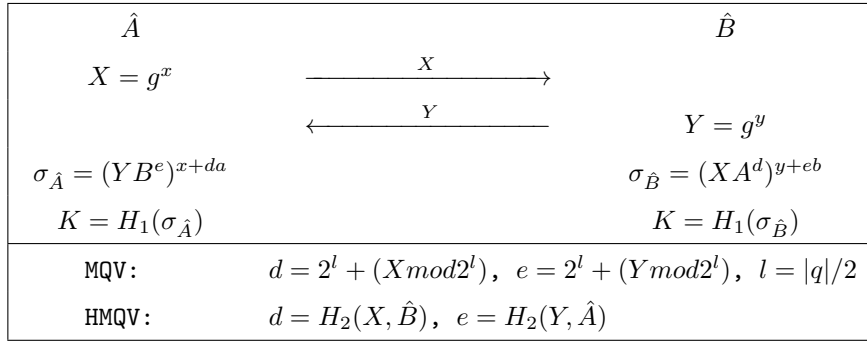


Figure 3: The MQV and HMQV protocols

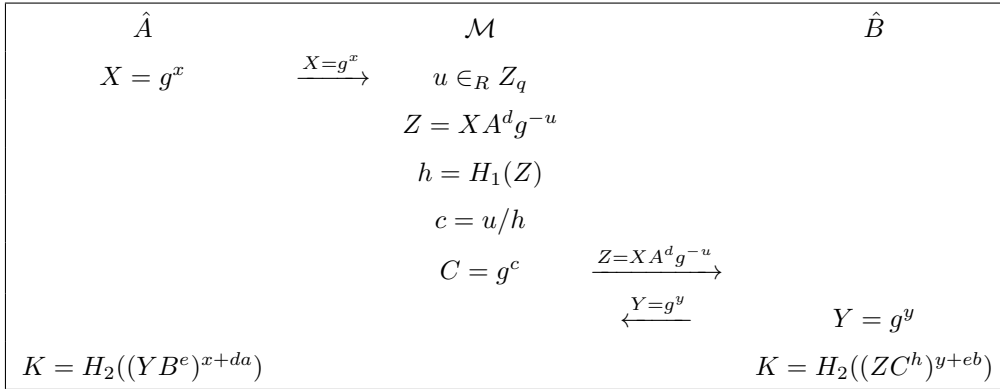


Figure 4: A UKS attack on MQV

as keys and crucial user data, in a shielded location where data is protected against interference and prying.

To operate the secret data in the shielded location, TCM provides a set of cryptographic APIs for users. Take the SM2-2 key exchange for example, TCM provides *TCM.CreateKeyExchange* and *TCM.GetKeyExchange* to generate a private/public key pair and generate a session key respectively:

- *TCM.CreateKeyExchange*: TCM generates a private/public SM2 key pair, which we denote by (a, g^a) , in the TCM’s shield location, and returns the public part of the SM2 key pair.
- *TCM.GetKeyExchange*: Input a public key of SM2, e.g, g^b , and return a session key g^{ab} .

TCM provides protection for cryptographic keys in the following two aspects. First, a user who controls an SM2-2 key pair generated by TCM cannot get plaintext of the private key, and the only way he can use the SM2-2 key is through TCM APIs. Second, as the key is randomly generated by TCM and the user has no control of the generation of a specific keys, a user cannot make TCM chips generate a specified key pair. The second protection feature constrains the adversary \mathcal{M} from using TCM to register a specified key.

3.1 Implementation of tKEA

Here we show how to implement KEA protocol using TCM. Our implementation consists of two phases: registration phase and key exchange phase.

The registration phase involves a security TCM chip \mathcal{T} , a Host \mathcal{H} , and a CA \mathcal{C} . \mathcal{T} and its host \mathcal{H} compose a whole entity. Before the registration phase, \mathcal{T} generates an attestation identity key (AIK) pair (sk_T, pk_T) (AIK is used to identify the platform in trusted computing, here we use it to certify the long-term key of an entity) and then registers the public key pk_T to a CA (note that this CA issues certificates to platforms, and is not the CA in the registration phase, which issues certificates to long-term keys) through protocols such as Privacy-CA [9], which is out of the scope of this paper. If higher anonymity is required, please refer to DAA [6] solution. After getting the AIK certificate, the registration proceeds as follows:

- 1) \mathcal{H} calls *TCM.CreateKeyExchange* command of \mathcal{T} , and \mathcal{T} generates an SM2-2 key pair (a, g^a) representing the long-term key of this entity.
- 2) \mathcal{H} then calls *TCM.CerifyKey* command of \mathcal{T} , and \mathcal{T} makes a **statement** about (a, g^a) using the AIK: “this key is held in a TCM-shielded location, and it will never be revealed”, and returns the **statement** to \mathcal{H} . The **statement** is actually a signature of the SM2-2 key by AIK. The AIK has a feature that it only signs the key generated within the TCM. This

feature assures the CA that the SM2-2 key is a real TCM-generated key if it has a legal signature.

- 3) \mathcal{H} transmits the **statement** to \mathcal{C} . \mathcal{C} verifies the **statement** to make sure that the public key g^a is generated by a real TCM chip. If the verification passes, \mathcal{C} issues a Cert about g^a and gives it to \mathcal{H} .

The key exchange phase is shown in Figure 5, and actually is the procedure of running the KEA protocol between two entities, e.g., \hat{A} and \hat{B} . \hat{A} consists of a TCM \mathcal{T}_1 and its host \mathcal{H}_1 , and \hat{B} consists of a TCM \mathcal{T}_2 and its host \mathcal{H}_2 . \hat{A} 's long-term public key is $A = g^a$, and \hat{B} 's long-term public key is $B = g^b$.

4 Security Model for tKEA

In this section we introduce a variant of CK model on which the security analysis of tKEA is based. For further details of CK model, please consult [7] for complete details. We modify the CK model by 1) modifying the *corruption(entity)* in the CK model, 2) adding an *establish(entity)* query to the queries of an adversary in the AKE experiment. The modified *corruption(entity)* query can simulate the protection of cryptographic keys provided by TCM, and the *establish(entity)* query allows an adversary to register public keys of adversary-controlled entities at any time in the experiment, that is, the adversary is allowed to mount the UKS attack.

4.1 Sessions

tKEA runs in a network of interconnected entities where each entity can be activated to run an instance of the protocol called a session. Within a session an entity can be activated to initiate the session or to respond to an incoming message. As a result of these activations, the entity creates and maintains a session state, generates outgoing messages, and eventually completes the session by outputting a session key and erasing the session state. There are two roles during a session, the entity that sends the first message in a session is called the **initiator** and the other the **responder**. We let \mathcal{I} denote initiator and \mathcal{R} denote responder. We identify an AKE session by a 5-tuple $(role, \hat{A}, \hat{B}, X, Y)$ where *role* denotes the role, X is the outgoing DH value and Y is the incoming DH value to the session. The session $(\mathcal{R}, \hat{B}, \hat{A}, Y, X)$ (if it exists) is said to be **matching** to session $(\mathcal{I}, \hat{A}, \hat{B}, X, Y)$.

4.2 Adversary

The AKE experiment involves multiple honest entities and an adversary \mathcal{M} connected via an unauthenticated network. The adversary \mathcal{M} is modeled as a probabilistic Turing machine and controls all communications. \mathcal{M} can intercept and modify messages sent over the network. \mathcal{M} also schedules all session activations and session-message

delivery. In addition, in order to model potential disclosure of secret information, the adversary is allowed to access secret information via the following queries:

- *session-state(s)*: \mathcal{M} queries directly at session s which is still incomplete and learns the session state for s . The session state may include, for example, the secret exponent of an ephemeral DH value but not the long-term secret key.
- *session-key(s)*: \mathcal{M} obtains the session key for a session s , provided that the session holds a key.
- *corruption(entity)*: For the information not stored in the TCM's shield location, such as the session states and session keys, \mathcal{M} learns all of them. For the long-term key stored in the TCM's shield location, \mathcal{M} has the ability to use it, such as computing $CDH(A, X)$ (A stands for the long-term public key, X stands for an element in G whose exponent is unknown) but cannot get the plaintext of the private key.
- *establish(entity)*: This query allows \mathcal{M} to register a public key generated in TCM, and \mathcal{M} has the ability to use the private key of the registered key. If \mathcal{M} registers a public key not generated in TCM, the CA will deny this registration after checking the AIK signature of the public key. \mathcal{M} can use this query to control an entity.

The adversary can make queries above to gain local information. We say that a completed session is "clean" if this session as well as its matching session (if it exists) is not subject to any of session-state, session-key, corruption queries.

Eventually \mathcal{M} should select a clean completed session, which is called a test session, and make query **Test(s)** and is given a challenge value C .

- *Test(s)*: Pick $b \xleftarrow{R} 0, 1$. If $b = 1$, provide \mathcal{M} with $C \leftarrow session-key(s)$; otherwise provide \mathcal{M} with C , which is a value r randomly chosen from the probability distribution of session keys.

Now \mathcal{M} can continue to make session-state, session-key, corruption and establish queries but is not allowed to expose the test nor any of the entities involved in the test session. At the end of its run, \mathcal{M} outputs a bit b' . We will refer to an adversary that is allowed the Test query as a **KE-adversary**.

Definition 1. An AKE protocol Π is called SK-secure if the following properties hold for any KE-adversary \mathcal{M} defined above:

- 1) when two uncorrupted entities complete matching sessions, they output the same key, and
- 2) the probability that \mathcal{M} correctly guesses the bit b (i.e., outputs $b' = b$) from the Test query is no more than $1/2$ plus a negligible fraction.

¹ X and Y are transmitted to \mathcal{T} by *TCM_GetKeyExchange*.

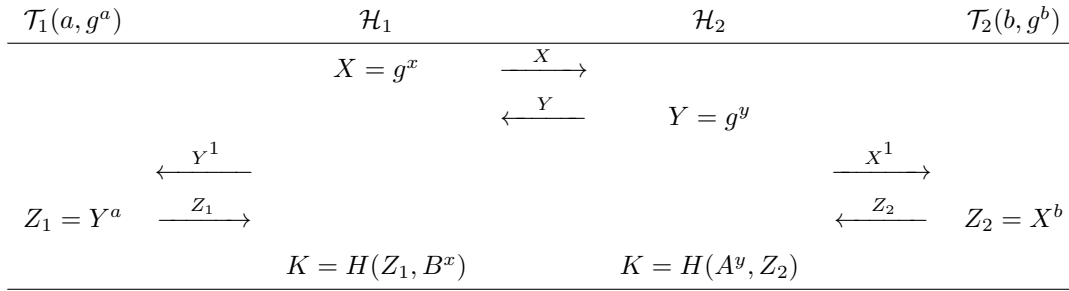


Figure 5: Implementation of KEA: tKEA

The advantage of **KE-adversary** participating in above AKE experiment against a protocol Π is defined as

$$\text{Adv}_{\Pi}^{\text{AKE}}(\mathcal{M}) = \Pr[\mathcal{M} \text{ wins}] - \frac{1}{2}.$$

5 Security of tKEA and MQV

5.1 Security Proof of tKEA

Under the *GDH* assumption in a group G and the protection provided by TCM chips, with the hash function $H()$ modeled as a random oracle, we show that tKEA satisfies AKE security against a **KE-adversary** defined in Section 4. The *GDH* assumption is that the CDH problem in G cannot be solved in polynomial time with non-negligible success probability even when a DDH oracle for G is available.

Let \mathcal{M} be any AKE adversary against tKEA. We start by observing that since the session key of the test session is computed as $K = H(\sigma)$ for some 2-tuple σ , the adversary \mathcal{M} has only two ways to distinguish K from a random value:

- 1) Forging attack. At some point \mathcal{M} queries H on the same 2-tuple σ as that of the test session.
- 2) Key-replication attack. \mathcal{M} succeeds in forcing the establishment of another session that has the same session key as the test session.

Let us first show that the key-replication attack is impossible if random oracles produce no collisions. If \mathcal{M} finds some session with the same 4-tuples as that of the test session, then this session must be executed by the same two entities, A and B . Let the ephemeral public keys of this session be X' and Y' . Since the session has the same signature as the test session, $CDH(A, Y')$ must be equal to $CDH(A, Y)$ and $CDH(B, X')$ - equal to $CDH(B, X)$. This implies that $X = X'$ and $Y = Y'$, and thus the session must be identical to the test session, which conflicts with the fact that the session is different from the test session.

However, the key-replication attack can happen to KEA. Lauter and Mityagin describe this attack in [21]. We here review this attack. An adversary \mathcal{M} registers a public key g^a of some honest entity \hat{A} as \mathcal{M} 's own public

key. Then \mathcal{M} intercepts a key-exchange session between \hat{A} and \hat{B} , and at the same time starts a session between \mathcal{M} and \hat{B} . \mathcal{M} forwards ephemeral public key g^x from \hat{A} to \hat{B} and ephemeral public key g^y from \hat{B} to \hat{A} . Since \mathcal{M} has the same public key as \hat{A} , both \hat{A} and \hat{B} will complete identical session keys, however they participate in two different sessions. \hat{B} participates in a session with \mathcal{M} while \hat{A} participates in a session with \hat{B} . Then \mathcal{M} can announce one of the two sessions as a test session and reveals the session key of the other session. To avoid UKS attacks, KEA+ adds the identities of the participating entities to the tuples, see Figure 2. This slight modification prevents adversaries to activate a session with the same tuple, thereby preventing \mathcal{M} from performing a key-replication attack. We show below that the protection provided by TCM can also prevent UKS attacks.

In the tKEA, we demonstrate that if an adversary \mathcal{M} plays a key-replication attack, he can break the protection provided by TCM. We denote the test session by s and the corresponding 2-tuple by $(CDH(A, Y), CDH(B, X))$. Correspondingly, we denote another session by s' which has the same session key with s , and the corresponding 2-tuple on which \mathcal{M} queries H to get the session key of s by $(CDH(A', Y'), CDH(B', X'))$. A' and B' are public keys \mathcal{M} registers to the CA through the *establish(entity)*, and \mathcal{M} can do the computation of $CDH(A' \text{ or } B', T)$ for any T whose exponent is unknown. Since s and s' has the same session key, $CDH(A', Y')$ must be equal to $Z_1 = CDH(A, Y)$ and $CDH(B', X')$ must be equal to $Z_2 = CDH(B, X)$. Since $CDH(A', Y') = Z_1$, we can get $Y' = Z_1^{\frac{1}{a'}}$ and $A' = Z_1^{\frac{1}{y'}}$. The only two ways for \mathcal{M} to get a pair (A', Y') meeting equation $CDH(A', Y') = Z_1$ are:

- 1) Register a controlled key A' to the CA, and compute the ephemeral public key $Y' = Z_1^{\frac{1}{a'}}$ where a' denotes the private key of A' .
- 2) Generate an ephemeral key pair $(y', Y' = g^{y'})$, and register $A' = Z_1^{\frac{1}{y'}}$ to the CA.

We can see that the first way requires \mathcal{M} to get the plaintext of the public key A' , and the second way requires \mathcal{M} to register a specified key. However, both of the two ways violate the protection provided by TCM which is described in Section 3.

We are left to show the impossibility of a forging attack. The proof of tKEA is similar to KEA+ [21]. It can be directly obtained by placing the 4-tuple of KEA+ on which is used to query H with tKEA's 2-tuple. So we omit the proof.

To summarize the proof, for any AKE adversary \mathcal{M} running in time t we can construct a GDH solver \mathcal{S} which runs in time $O(t^2)$ such that

$$\text{Adv}^{GDH}(\mathcal{S}) \geq \frac{1}{nk} \text{Adv}_{tKEA}^{AKE}(\mathcal{M})$$

As for the wPFS and KCI security property of tKEA, they can be proved directly following the proof in [21].

5.2 Securing MQV

To prove the generality of our way, we show that our way of using the protection capability provided by TCM/TPM to prevent UKS attacks can prevent the UKS attack [17] on MQV protocol. Figure 4 shows this attack. To attack MQV, the adversary \mathcal{M} registers an public key $C = g^c$ to the CA. As \mathcal{M} knows the private key of C , the CA cannot deny the registration of C even it require proof of knowledge of the private key. However, if the CA requires that the key must come from a security chip, such as TPM or TCM, this UKS attack can be prevented. That's because if the key is generated in a security chip, \mathcal{M} cannot generate a key whose private key is specified to be c . That's to say, \mathcal{M} cannot register $C = g^c$ to the CA.

6 Conclusion and Future Work

This paper summarizes two kinds of UKS attacks on the implicitly authenticated key exchange protocol and corresponding countermeasures to the two kinds of attacks. One of the countermeasure requires the CA to check the possession of the private key, which is unpractical, and the other countermeasure is to add the identity during the derivation of the session key, which modifies the original protocol. Motivated by the protection capability provided by security chips, we present a new way of preventing UKS attacks on AKE protocol.

We introduce the protection capability provided by hardware security chips and give a variant of CK model which covers UKS attacks. Through the security proof of tKEA in our variant model, we show that our new way of preventing UKS attacks is effective and have some advantages compared to existing countermeasures. We also show the generality of our new way by preventing the UKS attack on MQV protocol.

In Section 5, we show that our new way can prevent the UKS attack on MQV without a formal proof. In the future, we hope to implement a 'tMQV' using a hardware security chip like TCM, and give it a formal proof. We also hope to check whether the protection capability of hardware security chips can provide other advantages to AKE protocols.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (61602325, 61170304, 61472468, 61572331), the International Cooperation Program on Science and Technology (2011DFG13000), the Project of Beijing Municipal Science & Technology Commission (Z141100002014001), the Project of Construction of Innovative Teams and Teacher Career Development for Universities and Colleges Under Beijing Municipality (No.IDHT20150507), and the Scientific Research Base Development Program of the Beijing Municipal Commission of Education (TJSHG201310028014).

References

- [1] M. Abdalla, F. Benhamouda, and P. Mackenzie, "Security of the j-pake password-authenticated key exchange protocol," in *Security and Privacy*, pp. 571–587, 2015.
- [2] R. Amin and G. P. Biswas, "Cryptanalysis and design of a three-party authenticated key exchange protocol using smart card," *Arabian Journal for Science and Engineering*, vol. 40, no. 11, pp. 3135–3149, 2015.
- [3] R. Amin, S. K. H. Islam, G. P. Biswas, M. K. Khan, L. Lu, and N. Kumar, "Design of anonymity preserving three-factor authenticated key exchange protocol for wireless sensor network," *Computer Networks*, vol. 101, pp. 42–62, 2016.
- [4] M. Bellare and P. Rogaway, "Entity authentication and key distribution," in *Annual International Cryptology Conference*, pp. 232–249, 1993.
- [5] S. Blake-Wilson and A. Menezes, "Unknown key-share attacks on the station-to-station (STS) protocol," in *International Workshop on Public Key Cryptography*, pp. 154–170, 1999.
- [6] E. Brickell, J. Camenisch, and L. Chen, "Direct anonymous attestation," in *Proceedings of the 11th ACM conference on Computer and Communications Security*, pp. 132–145, 2004.
- [7] R. Canetti and H. Krawczyk, "Analysis of key-exchange protocols and their use for building secure channels," in *International Conference on the Theory and Applications of Cryptographic Techniques*, pp. 453–474, 2001.
- [8] R. Canetti and H. Krawczyk, "Security analysis of ikes signature-based key-exchange protocol," in *Annual International Cryptology Conference*, pp. 143–161, 2002.
- [9] L. Chen and B. Warinschi, "Security of the tcg privacy-ca solution," in *IEEE/IFIP 8th International Conference on Embedded and Ubiquitous Computing (EUC'10)*, pp. 609–616, 2010.
- [10] K.-K. R. Choo, C. Boyd, and Y. Hitchcock, "Examining indistinguishability-based proof models for

- key establishment protocols,” in *International Conference on the Theory and Application of Cryptology and Information Security*, pp. 585–604, 2005.
- [11] W. Diffie and M. Hellman, “New directions in cryptography,” *IEEE Transactions on Information Theory*, vol. 22, no. 6, pp. 644–654, 1976.
- [12] W. Diffie, P. C. Van Oorschot, and M. J. Wiener, “Authentication and authenticated key exchanges,” *Designs, Codes and Cryptography*, vol. 2, no. 2, pp. 107–125, 1992.
- [13] M. S. Farash, S. H. Islam, and M. S. Obaidat, “A provably secure and efficient two-party password-based explicit authenticated key exchange protocol resistance to password guessing attacks,” *Concurrency & Computation Practice & Experience*, vol. 27, no. 17, pp. 4897–4913, 2015.
- [14] A. Fujioka, K. Suzuki, K. Xagawa, and K. Yoneyama, “Strongly secure authenticated key exchange from factoring, codes, and lattices,” in *International Conference on Practice and Theory in Public Key Cryptography*, pp. 467–484, 2015.
- [15] S. H. Islam, “Design and analysis of a three party password-based authenticated key exchange protocol using extended chaotic maps,” *Information Sciences*, vol. 312(C), pp. 104–130, 2015.
- [16] ISO/IEC, *Entity Authentication Mechanisms - Part 3: Entity Authentication Using Asymmetric Techniques*, ISO/IEC IS 9798-3, 1993.
- [17] B. S. Kaliski Jr, “An unknown key-share attack on the mqv key agreement protocol,” *ACM Transactions on Information and System Security*, vol. 4, no. 3, pp. 275–288, 2001.
- [18] J. Katz, “Universally composable multi-party computation using tamper-proof hardware,” in *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pp. 115–128, 2007.
- [19] H. Krawczyk, “HMQV: A high-performance secure diffie-hellman protocol,” in *Annual International Cryptology Conference*, pp. 546–566, 2005.
- [20] B. LaMacchia, K. Lauter, and A. Mityagin, “Stronger security of authenticated key exchange,” in *International Conference on Provable Security*, pp. 1–16, 2007.
- [21] K. Lauter and A. Mityagin, “Security analysis of kea authenticated key exchange protocol,” in *International Workshop on Public Key Cryptography*, pp. 378–394, 2006.
- [22] L. Law, A. Menezes, M. Qu, J. Solinas, and S. Vanstone, “An efficient protocol for authenticated key agreement,” *Designs, Codes and Cryptography*, vol. 28, no. 2, pp. 119–134, 2003.
- [23] A. Menezes, M. Qu, and S. Vanstone, “Some new key agreement protocols providing mutual implicit authentication,” in *Second Workshop on Selected Areas in Cryptography (SAC’95)*, 1995.
- [24] A. Menezes and B. Ustaoglu, “On the importance of public-key validation in the mqv and hmqv key agreement protocols,” in *International Conference on Cryptology in India*, pp. 133–147, 2006.
- [25] National Institute of Standards and Technology, *Skipjack and KEA Algorithm Specifications, Ver. 2.0*, May 29, 1998.
- [26] Official of State Commercial Cryptography Administration, *Functionality and Interface Specification of Cryptographic Support Platform for Trusted Computing*, 2007.
- [27] R. Pecori and L. Veltri, “3AKEP: Triple-authenticated key exchange protocol for peer-to-peer voip applications,” *Computer Communications*, vol. 85, pp. 28–40, 2016.
- [28] V. Shoup, *On Formal Models for Secure Key Exchange*, Technical Report RZ 3120 (#93166), IBM, Apr. 19, 1999 .
- [29] Trusted Computing Group, *Trusted Platform Module Library Part 3: Architecture Family 2.0*, Jan. 7, 2014.
- [30] S. Zhao, L. Xi, Q. Zhang, Y. Qin, and D. Feng, “Security analysis of sm2 key exchange protocol in TPM2.0,” *Security and Communication Networks*, vol. 8, no. 3, pp. 383–395, 2015.
- [31] S. Zhao and Q. Zhang, “SHMQV: An efficient key exchange protocol for power-limited devices,” in *Information Security Practice and Experience*, pp. 154–167, 2015.
- [32] S. Zhao and Q. Zhang, “A unified security analysis of two-phase key exchange protocols in TPM 2.0,” in *International Conference on Trust and Trustworthy Computing*, pp. 40–57, 2015.
- [33] Q. Zhang, S. Zhao, Y. Qin, and D. Feng, “Improving the security of the hmqv protocol using tamper-proof hardware,” in *International Conference on Security and Privacy in Communication Systems*, pp. 343–361, 2014.

Biography

Qianying Zhang received her Ph.D degree from Institute of Software, Chinese Academy of Sciences in 2015. She is currently a lecturer in Capital Normal University. Her research interests include information security, operating system security, and formal verification.

Zhiping Shi received his Ph.D degree from Institute of Computing Technology, Chinese Academy of Sciences in 2005. He is currently an associate researcher in Capital Normal University. His research interests include formal verification, and artificial intelligence.

Whirlwind: A New Method to Attack Routing Protocol in Mobile Ad Hoc Network

Luong Thai Ngoc^{1,2}, Vo Thanh Tu¹

(Corresponding author: Luong Thai Ngoc)

Faculty of Information and Technology, Hue University of Sciences, Hue University, Viet Nam¹
77 Nguyen Hue street, Hue city, Vietnam

Faculty of Mathematics and Informatics Teacher Education, Dong Thap University, Viet Nam²
783 Pham Huu Lau street, Ward 6, Cao Lanh city, Dong Thap, Viet Nam

(Email: ltngoc@dthu.edu.vn)

(Received June 28, 2016; revised and accepted Nov. 15, 2016 & Jan. 11, 2017)

Abstract

Mobile Ad hoc Network (MANET) is a collection of wireless mobile nodes that dynamically create a network without a fixed infrastructure. However, all the characters make the security problem more serious, denial-of-Service attack is the main challenge in the security of MANET. In this article, we review some routing protocol attacks on Mobile Ad hoc Network. Specially, we propose a new attack method is called Whirlwind which originates one data Whirlwind on network that contain malicious node once the source node discovers a new route. And all data packets are resulted in drop due to over time-life without reaching the desired destination. We have, using the simulation system NS2, evaluated the harms of such attack on AODV protocol.

Keywords: AODV; MANET; Network Security; Routing Attacks

1 Introduction

Mobile Ad hoc Network is a special wireless, the advantages such as flexibility, mobility, resilience and independence of fixed infrastructure, nodes of the MANET network are coordinated with each other to communicate, data transfer among nodes is achieved by means of multiple hops. Hence, every mobile node acts both as a host and as a router [7].

Routing is the main service provided in network layer, the source node using the route to the destination is discovered and maintained. Routing protocols used in infrastructure networks cannot be applied in infrastructure-less networks like MANETs. Hence, many routing protocols are recommended to adapt to MANET, they are classified into proactive, reactive, and hybrid routing [1]. Proactive routing protocol is suitable with stable network topology because routes of network nodes must be estab-

lished to connect with other nodes before routing, typically DSDV [14], and OLSR [6]. In contrary, if network structure is regularly changed, then reactive routing is more suitable because nodes only discover routes in case of necessity by sending packet for route request and receiving packet for route answer, typically DSR [9], and AODV [15]. In the complex network topology, then typical routing protocols such as ZRP [5], and ZHLS [8] under the hybrid routing is more suitable to select.

Denial of service (DoS) attacks aim to deny a user of a service or a resource he would normally expect to have. Routing service at network layer is the target of many DoS [16], in which a malicious node will try to keep their resource but occupy other node's resource, for example, Blackhole [12], Sinkhole [3], Grayhole [4], and Flooding [17] under DoS attacks. Another way to interrupt routing service is to use a private tunnel connected between two malicious nodes. The result is that normal nodes will transfer data via this tunnel that appears the destination route with low cost. This type of attack is often called Wormhole [2, 10].

Ad hoc On-demand Distance Vector (AODV) is one of the most popular reactive routing protocol used for Ad hoc Networks. If source node N_S wants to communicate with destination node N_D without available route to destination, then N_S starts route discovery process by broadcasting the route request packet (RREQ) to destination. Destination node will answer to source about route by sending reply packet (RREP), maintain the route through HELLO and RERR packets. This is typical protocol under on-demand routing protocol, hence, hackers are easy to perform attacks on this protocol.

1.1 Blackhole/Sinkhole Attacks

Blackhole attack [12] is done by a malicious node or collaboration of harmful nodes. In the attack, a malicious node replies to source's RREQ packet by fake RREP (FR-

REP) packet with the best route to destination. By doing that, the Blackhole node successfully gains traffic flow from source transfer to destination. As result, the sources node sends all of data packets to the attack node which can drop or modify the packets. Another attack resemble Blackhole, called Sinkhole, was introduced in [3].

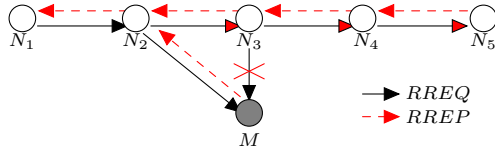


Figure 1: Description of blackhole attacks

In Figure 1, source node (N_1) discovers a new route to destination node (N_5) by broadcasting RREQ packet and then receive RREP packet. The best route from N_1 to N_5 on direction $\{N_1 \rightarrow N_2 \rightarrow N_3 \rightarrow N_4 \rightarrow N_5\}$ is established. However, the existing of a malicious node in the network N_2 establishes route to destination through malicious node M because M pretends it having the best route to N_5 by replying FRREP packet.

1.2 Grayhole Attacks

Grayhole attack [4] is similar to Blackhole attacks type, the destruction level is however less than, it also passes through 2 phases: *Phase 1*, malicious code shall self-advertise the source node that malicious node itself has route to destination with the lowest cost, it therefore can cheat the source node to change direction to destination through it. *Phase 2*, malicious node receives all packets from source and then drops the packet in different frequency, the malicious code sometime represents as normal node to prevent any detection. In order to advertise that it has route to destination with the lowest cost, the malicious node also uses FRREP packet as Blackhole attack.

1.3 Wormhole Attacks

They have described several types of Wormhole based on the techniques used to tunnel the packets between the colluding nodes, such as: Wormhole through the tunnel (called out-of-band channel - OB), Wormhole using encapsulation, Wormhole using packet relay, Wormhole with high power transmission [11]. Especially, authors [10] described that all of them may be operated for two modes of attacks: Hidden Mode (HM) and Participation Mode (PM). In HM, malicious nodes are hidden from normal nodes, when receive packets and simply forwards them to each other without process packet, thus, they never appear in routing tables of neighbors. In contrast, PM malicious nodes are visible during the routing process because they processes packets as normal nodes. Note that the malicious node appears in routing tables of neighbors and the HC increase when packet is forwarded.

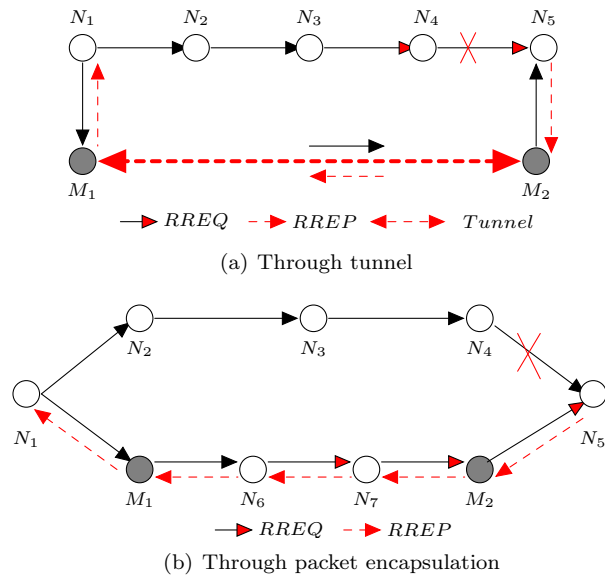


Figure 2: Description of wormhole attacks types

Out-of-band channel: Attacker using 2 malicious nodes connect to each other and create a private channel called “tunnel” aiming to minimize hop count (HC) when source node discovers route by RREQ packet. In Figure 2(a), source node N_1 requests the route to destination N_5 by broadcasting RREQ via 2 routes $\{N_1 \rightarrow N_2 \rightarrow N_3 \rightarrow N_4 \rightarrow N_5\}$ and $\{N_1 \rightarrow M_1 \rightarrow M_2 \rightarrow N_5\}$. Finally the second route through M_1, M_2 was established because it has the best traffic cost.

Encapsulation: To attack, malicious nodes (M_1, M_2) appear in the network similar to normal nodes. When M_1 received the RREQ packet, which encapsulates it and forwards it to M_2 via normal nodes. Node M_2 is responsible for decapsulation the packet before send it to destination. Because of the packets encapsulation, the routing cost not increase during the traversal through the normal nodes. As a result, source node discovers a new route which contains malicious node. In Figure 2(b), source node broadcasts RREQ packet to destination node N_5 following to 2 routes $\{N_1 \rightarrow N_2 \rightarrow N_3 \rightarrow N_4 \rightarrow N_5\}$ and $\{N_1 \rightarrow M_1 \rightarrow N_6 \rightarrow N_7 \rightarrow M_2 \rightarrow N_5\}$. When M_1 received the RREQ packet, it encapsulates the packet then forward RREQ into current route. M_2 node is responsible for decapsulation the packet before broadcast it to N_5 . The same process also happens when RREP generated by N_5 forwarding back to N_1 through M_2 and M_1 . The purpose is keeping HC not increase while the packets travel from M_1 to M_2 and vice versa. As a result, the RREP from N_5 follow the second route is better than others, hence N_1 obviously chooses the route to N_5 through two malicious nodes.

Using packet relay: The main idea is a malicious node relays fake packets between two non-neighbor nodes creating an illusion that they are neighbors, the purpose is insert itself into route.

Wormhole with high power transmission:

Malicious node has a high power antenna, thus distant nodes receive the RREQ packet faster from the malicious node. The result discovered route may contain malicious node because its routing cost is cheaper normal route.

1.4 Flooding Attacks

Flooding attack [17] is one of the main challenges in the security of MANETs. It is implemented by overwhelmingly sending control packets or useless data packets from malicious nodes to unavailable destinations. The result is a broadcasting storm of packets and increasing communication overhead, which reduce the responsiveness at each node because of its unnecessary processing of the flooded packets. For AODV, Flooding attacks try to send HELLO, RREQ and DATA packets at a high frequency.

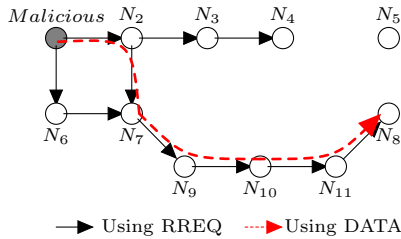


Figure 3: Description of flooding attacks

HELLO packet flooding: In MANETs, nodes periodically broadcast HELLO packets to notice their existence with their neighbors. A malicious node abuses this feature to broadcast HELLO packets at a high frequency that force its neighbor nodes to spend their resources on processing unnecessary packets. This HELLO packet flooding is only detrimental to the neighbors of a malicious node. See in Figure 3, both nodes N_2 and N_6 are affected by malicious node N_1 .

RREQ packet flooding: In AODV protocol, nodes broadcast RREQ packet to discover routes. To attack, a malicious node continuously and excessively broadcasts RREQ packets, which causes a broadcast storm in the network and floods with unnecessary packets being forwarded. The RREQ flooding attack is seen as the most harmful because it has a great impact on the route discovery in the network. It also causes high resource consumption at affected nodes and increases the communication overhead. See in Figure 3, all nodes in topology are affected by malicious node N_1 .

DATA packet flooding: A malicious node can excessively broadcast data packets to any nodes in the network. This can waste other nodes' resources and bandwidth. It can create congestions in the network. This kind of attack has more impact on the nodes participating in the data routing to the destinations. Figure 3, DATA packet flooding attacks effects all node in route $\{N_2 \rightarrow N_7 \rightarrow N_9 \rightarrow N_{10} \rightarrow N_{11} \rightarrow N_8\}$.

2 Proposing Whirlwind Attack in Mobile Ad Hoc Network

2.1 Main Idea

Routing protocol is responsible for exploring the route to destination when source node wants to communicate. A good protocol is not a quick gather, low routing explore cost only, but being able to prevent routing loop is also an extremely important factor. Whirlwind attacks target is to make routing loop which is done with two phases:

Phase 1: Malicious node try to set up a routing loop path in the route from source to destination node when receiving RREQ packet from any source node N_S by using the FRREP packet. The detail process is showed in Algorithm 1.

Phase 2: If attacking is successful, all data packets from source N_S to destination node are taken into data whirlwind and automatically dropped due to over time-life. We have, basing on this feature, named this attack method as Whirlwind attacks.

2.2 Description of Whirlwind Attacks in AODV Protocol

AODV protocol uses the route exploration mechanism if it is necessary. If source node N_S wants to communicate with destination node N_D however route to destination is unavailable, N_S starts the route exploration process by broadcasting RREQ packet to destination node. Destination node replies route to source by sending unicast RREP packet. In AODV, all nodes remain route by using HELLO packet and update route by using RERR packet.

In normal network topology (Figure 4(a)), source node N_1 discovers route to destination node N_5 by broadcasting of RREQ to its neighbor nodes named N_2 . Intermediate node N_2 is not destination node, it therefore continue broadcasts RREQ packet to its neighbors named N_3 and save reserve route to source N_1 , this process repeats at N_3 and N_4 until node N_5 receives the route request packet.

When receiving RREQ packet from node N_4 , destination node N_5 sends unicast of RREP packet to source on route $\{N_5 \rightarrow N_4 \rightarrow N_3 \rightarrow N_2 \rightarrow N_1\}$. As a result, source node N_1 discovers route to destination in following direction $\{N_1 \rightarrow N_2 \rightarrow N_3 \rightarrow N_4 \rightarrow N_5\}$. The detail

Algorithm 1 Description of the process to set up a routing loop path in Whirlwind attacks

- 1: Begin
- 2: *Step 1:* Malicious node M wait until receiving the RREQ packet from source node N_S ;
- 3: *Step 2:* When receiving the first RREQ packet from node N_i , node M adds route to destination N_D via N_i into its routing table (RT); and waiting to receive the second RREQ packet;
- 4: *Step 3:* When receiving the second RREQ of N_S from node N_j , malicious node M adds route to source N_S via N_j into its RT; and sends FRREP packet to source node via next hop (NH) N_j to inform N_j about M with route to destination N_D with lowest cost and “fresh” enough;
- 5: *Step 4:* If M does not receive the second RREQ packet from N_S then this process is fail and the end;
- 6: *Step 5:* When receiving FRREP packet, N_j adds route to destination N_D through next hop M because it assumes that M has route to destination with minimum cost;
- 7: *Step 6:* The FRREP packet is forwarded by N_j to source node through revert route (recorded in broadcast RREQ process) until source node receives FRREP packet;
- 8: *Step 7:* Destination node N_5 also replies to source node of RREP packet. Thus, source node receives two routing replies packet. However, the FRREP packet from malicious node is accepted due to it has lower cost and more “fresh”. The result is the route from source N_S to destination N_D has circle consisting of nodes named N_i , N_j and M ;
- 9: End

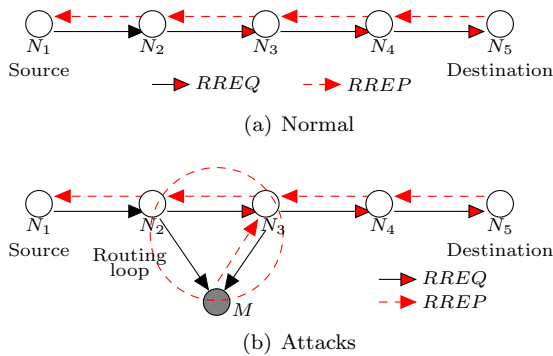


Figure 4: Description of route discovery in AODV

information of RREQ, RREP packets and routing table of each node are detailed in Table 1.

Figure 4(b) shows that malicious node M appears in network topology and conducting Whirlwind attack, M is neighbor of both nodes N_2 and N_3 . When receiving the first RREQ packet from node N_2 , malicious node M saves route to destination into its RT with minimum cost

Table 1: Results of discovery route in normal topology; Des: Destination address; Src: Source address

Steps	Nodes	RREQ/RREP (Src, Des, HC)	Routing Table		
			Des	NH	HC
RREQ	N_1	$N_1, N_5, 0$	NULL		
	N_2	$N_1, N_5, 1$	N_1	N_1	1
	N_3	$N_1, N_5, 2$	N_1	N_2	2
	N_4	$N_1, N_5, 3$	N_1	N_3	3
	N_5	$N_1, N_5, 4$	N_1	N_4	4
RREP	N_5	Creates RREP packet [$N_5, N_1, 0$]			
	N_4	$N_5, N_1, 1$	N_5	N_5	1
	N_3	$N_5, N_1, 2$	N_5	N_4	2
	N_2	$N_5, N_1, 3$	N_5	N_3	3
	N_1	$N_5, N_1, 4$	N_5	N_2	4

[$Des = N_5, NH = N_2, HC = 1$]. When receiving the second RREQ packet from node N_3 , malicious node saves the reserve route to source N_1 into its RT with lowest cost [$Des = N_1, NH = N_3, HC = 1$], concurrently sends unicast of FRREP to source N_1 in direction $\{M \rightarrow N_3 \rightarrow N_2 \rightarrow N_1\}$. As a result, routing table of node N_3 has route information to destination N_5 via NH is M with the cost of 1.

Destination node N_5 also replies to source node of RREP packet in direction $\{N_5 \rightarrow N_4 \rightarrow N_3 \rightarrow N_2 \rightarrow N_1\}$. When receiving the RREP packet from node N_4 , node N_3 see that the cost to destination N_5 is not cheaper than the existing route, the RREP packet is therefore dropped. Table 2 shows that exist routing loop on route from N_1 to N_5 in RT of nodes named N_2 , N_3 , and M . Therefore, malicious node M has successfully attacked.

Table 2: Results of discovery route in attacks topology

Steps	Nodes	RREQ/RREP (Src, Des, HC)	Routing Table		
			Des	NH	HC
RREQ	N_1	$N_1, N_5, 0$	NULL		
	N_2	$N_1, N_5, 1$	N_1	N_1	1
	M	$N_1, N_5, 2$	N_5	N_2	1 *
	N_3	$N_1, N_5, 2$	N_1	N_2	2
	M	$N_1, N_5, 3$	N_1	N_3	1 *
	N_4	$N_1, N_5, 3$	N_1	N_3	3
FRREP	N_5	$N_1, N_5, 4$	N_1	N_4	4
	M	Creates RREP packet [$N_5, N_1, 0$]			
	N_3	$N_5, N_1, 1$	N_5	M	1
	N_2	$N_5, N_1, 2$	N_5	N_3	2
RREP	N_1	$N_5, N_1, 3$	N_5	N_2	3
	N_5	Creates RREP packet [$N_5, N_1, 0$]			
	N_4	$N_5, N_1, 1$	N_5	N_5	1
	N_3	Drops RREP packet			

(*) Entry is added by malicious node

However, algorithm 1 shows that Whirlwind attack is done successful if malicious nodes receive full two RREQ packets from neighbors. In Figure 5 shows that Whirlwind attack is fail due to malicious node receive only one RREQ packet from N_2 .

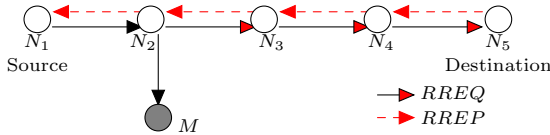


Figure 5: Whirlwind attacks is fail

2.3 Comparison of Whirlwind and Other Attacks

The recent studies show that hacker can perform many network attacks in MANET [16]. They can classify under some criteria named purpose, location, form, and lost packet cause. Attack purpose contain attack for dropping data and eavesdropping; and attack position convey external and internal; and attack forms consist of active and passive. Whirlwind attack also aims at dropping data, however data is dropped at normal node due to over time-life, this is differ from Blackhole and Grayhole attacks that packet is dropped by malicious node. Whirlwind is active attacks form, it is performed from internal location of network. Table 3 shows comparison of Whirlwind and other attacks.

Table 3: Summarized attack methods

Features		Attack types				
		BH	GH	WH	FD	WW
Purpose	Dropping	●	●	○	●	●
	Eavesdropping			●		
Localtion	External	●	●	●	●	
	Internal					●
Form	Active	●	●	●	●	●
	Passive		○			
Lost packets	Malicious nodes	●	●	●	●	
	Over time-life					●

(●) Implement; (○) Optional; BH: Blackhole; GH: Grayhole; WH: Wormhole; FD: Flooding; WW: Whirlwind;

3 Result Evaluation by Simulation

We evaluate the impact of Whirlwind attack on simulation system is NS2 [13] (version 2.35) on AODV protocol.

3.1 Simulation Settings

At the physical and data link layer IEEE 802.11 is used, the traffic pattern was generated using CBR as the data source and UDP protocol is used for transporting the data and the packet size is of 512 bytes, 200s of simulation; the transmission range of a node is 250m, FIFO queue (See more in Table 4).

Table 4: Simulation parameters

Parameters	Setting
Simulation time (s)	200
Wireless standard	IEEE 802.11
Ratio range (m)	250
Traffic type	CBR
Packet size	512 bytes
Queue type	FIFO (DropTail)

We used two network topology, (a) *Topology 1* is available with 5 normal nodes, using one CBR as the data source for transporting the data, 1 malicious node is immobile at the position as Figure 4. (b) *Topology 2* is available with 100 normal nodes and 1 malicious node, and operated in the area of 2000m x 2000m, malicious node is immobile at the central position, all nodes stay in Grid network topology as Figure 6, 10 data source CBR, the first CBR source is started at second of 0, the following CBR is 5 seconds apart from each source.

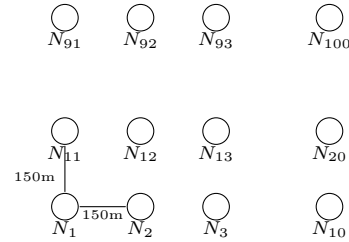


Figure 6: Grid network topology

3.2 Simulation Results

To evaluate the impact of Whirlwind attack, we use two criterion: *Packet delivery ratio and Network throughput*.

- 1) Packet delivery ratio (PDR): It can be measured as the ratio of the received packets by the destination nodes to the packets sent by the source node. $PDR = (\text{number of received packets} / \text{number of sent packets}) * 100$;
- 2) Network throughput: is the parameter of measuring information transported which is calculated by $(\text{total packet sent successfully} * \text{size of packet}) / \text{simulation time}$.

Packet delivery ratio: Figure 7 shown that Whirlwind attack had caused impact on route discovery ability of source node, hence the ratio of sending packet successfully has much been reduced. After finishing 200s simulation in the first network topology, the packet delivery ratio of AODV is 98.04% under normal network topology and there are nothing any

packet is sent to destination under Whirlwind attack. In Gird network topology, the packet delivery ratio of AODV is 97.47% under normal network topology and 31.97% under Whirlwind attack, 65.5% reduced.

Network throughput: Figure 8 shown that Whirlwind attack has reduced network throughput. After finishing 200s simulation, if one malicious node attacks, the network throughput of AODV is 0 bps in the first network topology. In Gird topology, throughput is 33,157.12 bps without attacks and 10,874.88 bps under Whirlwind attack.

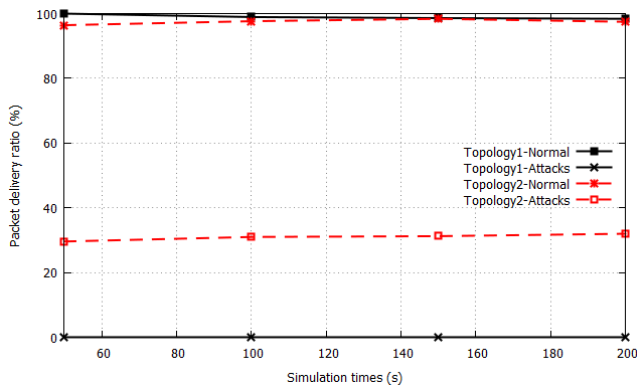


Figure 7: Packet delivery ratio

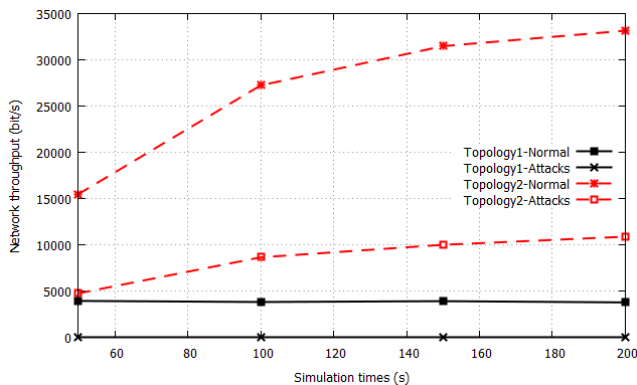


Figure 8: Network throughput

4 Conclusion

This article proposes a new attack method named Whirlwind that cause harm to performance of Mobile Ad hoc Network. Simulation results on AODV protocol show that the malicious node has successfully created data packet whirl-wind on network that cause loss packet, this decreases the packet delivery ratio, and network throughput. In Gird network topology, the packet delivery ratio of AODV 65.5% reduced under Whirlwind attack.

In the future, we shall continue installing the malicious assessment compared to other attacks on some routing protocols named DSR to evaluate harms.

References

- [1] E. Alotaibi and B. Mukherjee, "A survey on routing algorithms for Wireless Ad-hoc and Mesh networks," *Computer Networks*, vol. 56, no. 2, pp. 940–965, 2012.
- [2] A. P. Asad, C. McDonald, "Detecting and evading wormholes in mobile ad-hoc wireless networks," *International Journal of Network Security*, vol. 3, no. 2, pp. 191–202, 2006.
- [3] L. S. Casado, G. M. Fernandez, P. Garca-Teodoro, and N. Aschenbruck, "Identification of contamination zones for sinkhole detection in MANETs," *Journal of Network and Computer Applications*, vol. 54, pp. 62–77, 2015.
- [4] X. Gao and W. Chen, "A novel gray hole attack detection scheme for mobile ad-hoc networks," *IFIP International Conference on Network and Parallel Computing Workshops*, pp. 209–214, 2007.
- [5] Z. J. Haas, M. Pearlman, *The Zone Routing Protocol (ZRP) for Ad-Hoc Networks*, IETF Internet Draft, draft-ietf-manet-zone-zrp-04.txt, 2002.
- [6] P. Jacquet, P. Muhlethaler, T. Clausen, A. Laouiti, A. Qayyum, and L. Viennot, "Optimized link state routing protocol for ad hoc networks," in *IEEE International Multi Topic Conference*, pp. 62–68, 2001.
- [7] H. Jeroen, M. Ingrid, D. Bart, and D. Piet, "An overview of Mobile Ad hoc Networks: Applications and challenges," *Journal of the Communications Network*, vol. 3, no. 3, pp. 60–66, 2004.
- [8] M. Joa-Ng and I. T. Lu, "A peer-to-peer zone-based two-level link state routing for mobile ad hoc networks," *IEEE Journal on Selected Areas in Communications*, vol. 17, no. 8, pp. 1415–1425, 1999.
- [9] D. B. Johnson and D. A. Maltz, "Dynamic source routing in ad hoc wireless networks," in *Mobile Computing*, vol. 353, pp. 153–181, 1996.
- [10] J. Karlsson, L. S. Dooley, G. Pulkkis, "A new MANET wormhole detection algorithm based on traversal time and hop count analysis," *Sensors*, vol. 11, pp. 11122–11140, 2011.
- [11] M. Kumar, K. Dutta, I. Chopra, "Impact of wormhole attack on data aggregation in hierarchical WSN," *International Journal of Electronics and Information Engineering*, vol. 1, no. 2, pp. 70–77, 2014.
- [12] S. Kurosawa, H. Nakayama, N. Kato, A. Jamalipour, and Y. Nemoto, "Detecting blackhole attack on AODV-based mobile ad hoc networks by dynamic learning method," *International Journal of Network Security*, vol. 5, no. 3, pp. 338–346, 2007.
- [13] S. McCanne, S. Floyd, *The Network Simulator NS2*, Mar. 28, 2017. (<http://www.isi.edu/nsnam/ns/>)

- [14] C. E. Perkins, P. Bhagwat, "Highly dynamic destination sequenced distance-vector routing (DSDV) for mobile computers," *ACM SIGCOMM Computer Communication Review*, vol. 24, no. 4, pp. 234–244, 1994.
- [15] C. E. Perkins, M. Park, and E. M. Royer, "Ad-hoc on-demand distance vector routing," in *Proceedings of Second IEEE Workshop on Mobile Computing Systems and Applications (WMCSA'99)*, pp. 90–100, 1999.
- [16] R. Di Pietro, S. Guarino, N. V. Verde, and J. Domingo-Ferrer, "Security in wireless ad-hoc networks - A survey," *Computer Communications*, vol. 51, pp. 1–20, 2014.
- [17] Y. Ping, D. Zhoulin, Y. Zhong, and Z. Shiyong, "Resisting flooding attacks in ad hoc networks," in *International Conference on Information Technology: Coding and Computing (ITCC'05)*, vol. 2, pp. 657–662, 2005.

Biography

Luong Thai Ngoc is working in the Faculty of Mathematics and Informatics Teacher Education, Dong Thap University. He received B.E. degree in Computer Science from Dong Thap University in 2007 and M.A. degree in Computer Science from Hue University of Sciences in 2014. He is a PhD student in Hue University of Sciences now. His fields of interesting are network routing, analysis and evaluation of network performance, security wireless ad hoc network.

Vo Thanh Tu is an associate professor in the Faculty of Information Technology, Hue University of Sciences, Hue University. He received B.E. degree in Physics from Hue University in 1987 and PhD degree in computer science from Institute of Information Technology, Vietnam Academy of Science and Technology in 2005. His fields of interesting are network routing, analysis and evaluation of network performance, security wireless ad hoc network.

Distinguishing Medical Web Pages from Pornographic Ones: An Efficient Pornography Websites Filtering Method

Jyh-Jian Sheu

(Corresponding author: Jyh-Jian Sheu)

College of Communication, National Chengchi University

Taipei City 11605, Taiwan (R.O.C.)

(Email: jjsheu@nccu.edu.tw)

(Received Dec. 12, 2016; revised and accepted Feb. 19, 2017)

Abstract

In this paper, we apply the uncomplicated decision tree data mining algorithm to find association rules about pornographic and medical web pages. On the basis of these association rules, we propose a systematized method of filtering pornographic websites with the following major superiorities: 1) Check only contexts of web pages without scanning pictures to avoid the low operating efficiency in analyzing photographs. Moreover, the error rate is lowered and the accuracy of filtering is enhanced simultaneously. 2) While filtering the pornographic web pages accurately, the misjudgments of identifying medical web pages as pornographic ones will be reduced effectively. 3) A re-learning mechanism is designed to improve our filtering method incrementally. Therefore, the revision information learned from the misjudged web pages can incrementally give feedback to our method and improve its effectiveness. The experimental results showed that each efficacy assessment indexes reached a satisfactory value. Therefore, we can conclude that the proposed method is possessed of outstanding performance and effectivity.

Keywords: Data Mining; Decision Tree; Medical Web Page; Pornographic Websites Filtering

1 Introduction

Given the anonymity of the Internet, the number of pornographic websites has been increasing steadily in the past decade. The overflow of pornographic information on the Internet has not only imposed impacts on the mental and physical health and values of children or youngsters, but also included by the scholars as one of the causes of physical and mental damage [21]. There are various mechanisms of filtering pornographic websites at present. We study and organize the prevailing filtering methods as

the following four categories:

- 1) Website rating: This method is to filter the web pages by applying rating (or classification) tags [6, 15, 18]. However, this method is lacking in that it is reliant on the initiatives of the website builders. Without any mandatory force, the implementation could not always meet the desired filtering effects.
- 2) Static filtering: This method works to establish a blacklist of pornographic websites that should be forbidden through website URL, DNS, or the ports of TCP/IP protocols [5, 8, 9]. There are two major types: Site blocking and Internet service blocking. Since the method does not involve the analysis of website contents, there is a high chance that it would make wrong judgments concerning normal websites.
- 3) Dynamic filtering: It determines whether a website is pornographic or not by analyzing the website content. The analysis on the content and features of website is usually conducted via algorithms, with the aim to discover relevant rules. Dynamic filtering can be divided into two categories: keyword filtering and intelligent content analysis [1, 23].
- 4) Images filtering: This mechanism would first determine whether what the image represents are human limbs via edge detection and then decide whether the connected groups of limbs could constitute a human figure [3]. In recent years, a lot of filtering methods of pornographic images have been proposed [11, 13, 24, 26]. Moreover, combined with various techniques, numerous intelligent methods of filtering pornographic websites are proposed in succession [4, 7, 12, 27]. However, the excessive computation of scanning images might bring about the low operating efficiency.

Some websites, such as those featuring medical, physical educational and artistic themes, tended to be eas-

ily suspected as phishing websites during the detection process. According to the survey report issued by Pew/Internet, there are about 100,000 medical / health websites around the world, among which over 10,000 are set in the United States [17]. However, among the information on these medical and health care websites, general knowledge on health care, professional information concerning diseases, beauty & slimming, and other health and mental information like sexual knowledge (relevant medicines, methods of birth control, treatment of venereal diseases, etc.); information concerning special periods (pregnancy, parenting, maintenance and physique improvement during puberty); individual mental and health care sharing (fighting pressure, discussions) etc., tend to incorporate pornographic keywords in their contents. For example, the website of American corporative Planned Parenthood Federation¹, this is a legal web page on medical education. But given the existence of suspected pornographic keywords, this web page might be judged as a suspected pornographic web page in spite of its legitimacy.

This study aims to present an efficient mechanism of filtering pornographic websites based on the machine learning technique. In this paper, we apply the uncomplicated decision tree data mining algorithm to find association rules about pornographic web pages. On the basis of these association rules, we propose a systematized method of filtering pornographic websites with the following major superiorities:

- 1) In order to avoid the low operating efficiency in analyzing photographs, we check only contexts of web pages without scanning pictures. Moreover, the error rates (classify a pornographic website as non-pornographic or a non-pornographic website as pornographic) will be lowered and the accuracy of filtering will be enhanced simultaneously.
- 2) While filtering the pornographic web pages accurately, the misjudgments of identifying medical web pages as pornographic ones will be reduced effectively.
- 3) A re-learning mechanism is designed to improve our filtering method incrementally. We apply a supervised machine learning skill to collect any pornographic keywords found newly in the misjudged web pages. Therefore, the revision information learned from the misjudged web pages can incrementally give feedback to our method and improve its effectiveness.

The remainder of this paper is organized as follows. Section 2 introduces the decision tree data mining algorithm. The detailed architecture of our filtering method is shown in Section 3. In Section 4, the experimental results of our method will be presented. Section 5 concludes this paper.

2 Decision Tree Data Mining Algorithm

In this paper, based on decision tree data mining technique, we will propose an efficient systematized method of filtering pornographic websites. The proposed method will analyze the association rules about pornographic web pages and apply them to classify the unknown web pages to be either pornographic or non-pornographic. Decision tree is one of the widely used data mining methods. The technology excels in that it could generate easily understandable association rules and visual features via easy calculations.

There are various decision tree algorithms. Iterative Dichotomiser 3 (called ID3 for short) proposed by Quinlan is one of the most famous and effective decision tree algorithms [19, 20]. According to the study of Stark and Pfeiffer [22], the behavior of ID3 would be better than other improved versions, such as C4.5, CHAID, and CART. Ohmann et al. demonstrated that the quantity of association rules produced by ID3 would not be as numerous as that of C4.5 [16]. Hence, they concluded that ID3 algorithm possessed the superior feature because of the simplicity of rule quantity. Therefore, we adapt ID3 as the data mining technique in this study.

A decision tree is made of a start node (called root node), leaf nodes, and the internal nodes (also called non-leaf node) between the root node and the leaf nodes. In the tree structure, the upper node (called parent node) might branch downward some adjacent nodes (called children nodes). And the final nodes without any branch are called leaf nodes. Suppose that "Target Attribute" is the attribute which is concerned objective of our research. For example, the attribute "web type" ("P" means pornographic websites; "M" means medical websites; "N" means normal websites) is the Target Attribute in this study. Moreover, let "Critical Attributes" be the other important attributes of web pages. The building of the decision tree starts from the root node when all the data instances are contained in the root. The ID3 algorithm will pick out the Critical Attribute with the highest "Information Gain", according to whose values the data instances within the node will be divided into different children nodes. The same process will be repeated by the children nodes on their respective data instances. The algorithm will be ended under two conditions: 1) repeated until all the critical attributes have been selected; or 2) there will be no need of further divisions if the values of the Target Attribute concerning all the data instances within the node are the same. When either of these conditions is met, the current node will be marked as a leaf node. This leaf node will be labeled as the value of Target Attribute possessed by the majority of data instances in this node. Then the algorithm will be stopped and the decision tree is generated completely.

Given a leaf node C, we assume that the value of Target Attribute possessed by the majority of data instances

¹<http://www.plannedparenthood.org/>

in C is denoted as $Label(C)$, and $|LabelC|$ is the number of data instances whose Target Attribute's value is the same as $Label(C)$ in C . Then we will compute C 's degree of purity (denoted as $Purity(C)$) and degree of support (denoted as $Support(C)$) for this leaf node C . The formulas of $Purity(C)$ and $Support(C)$ are defined as follows:

$$\begin{aligned} Purity(C) &= (|Label(C)|/|C|) \times 100\% \\ Support(C) &= (|C|/N) \times 100\% \end{aligned}$$

where $|C|$ is the number of data instances contained in node C and N is the number of total data instances.

In the resulted decision tree, each path from the root node to a leaf node constitutes an association rule. That's to say, all the internal nodes along the path will serve as the "if" condition for the series of Critical Attributes, together with the "then" outcome represented by the labelled Target Attribute's value of the leaf node, an "if-then" association rule is thus formed.

As compiled by this study, the major calculation steps of ID 3 algorithm are as follows [19]:

- 1) The algorithm begins from the root node C , when all the data instances are contained in C .
- 2) If all the data instances within node C have the same value of Target Attribute, then define it as a leaf node, label C by this value, compute $Purity(C)$ and $Support(C)$, and end the algorithm. Otherwise, move on to next step.
- 3) If all the Critical Attributes have been selected, the values of the Target Attribute concerning the data instances within node C should be examined via majority voting, thus picking out the value boasting the largest number of data instances. Then node C should be defined as a leaf node and labelled by this value, thus computing $Purity(C)$ and $Support(C)$ and ending the algorithm. Otherwise, move on to next step.
- 4) Calculate the Entropy $E(C)$ for node C through the following expression:

$$E(C) = - \sum_{i=1}^t P_i \times \log_2 P_i$$

where t is the number of Target Attribute's values, and $P_i = (\text{the number of data instances whose values of the Target Attribute corresponding to the } i^{th} \text{ value, } 1 \leq i \leq t, \text{ in } C) / (\text{the total number of data instances in } C)$.

- 5) For each Critical Attribute that has not been selected yet (assumed to be attribute α), the Entropy $E^+(\alpha)$ and Information Gain $G(\alpha)$ will be computed by the following expressions respectively:

$$\begin{aligned} E^+(\alpha) &= \sum_{j=1}^k (n_j/n) \times E(C_j); \\ G(\alpha) &= E(C) - E^+(\alpha). \end{aligned}$$

In the expressions, we assume that attribute α is supposed to have k values, C_j (for $1 \leq j \leq k$) represents the subset of the data instances whose values concerning attribute are the same, $E(C_j)$ refers to the Entropy of the subset as calculated through the equation in Step (4), n stands for the total number of data instances within C , and n_j represents the total number of data patterns in the subset C_j .

- 6) Choose the Critical Attribute that has not been selected yet boasting the highest Information Gain. Assume that the selected Critical Attribute has m values, the children nodes C_1, C_2, \dots, C_m should be built under this node, to which the data instances of node C should be distributed according to their values of the select Critical Attribute.
- 7) Respectively treat every children node C_i as node C , $1 \leq i \leq m$, continue the algorithm recursively from Step (2).

3 An Efficient Pornographic Websites Filtering Mechanism

The objective of this research is to filter pornographic web pages, namely, judging an unknown web page as either pornographic or non-pornographic. While filtering the pornographic web pages, great efforts have been taken to avoid misjudging medical web pages as pornographic ones. For this purpose, medical web pages are set apart from normal web pages in its own category.

In this research, we propose a three-phase systematic method of filtering pornographic websites by applying ID3 decision tree algorithm. The proposed method is possessed of the ability to discriminate between pornographic websites and medical website. Assume that websites will be classified into three categories: "pornographic", "medical", and "normal". Based on the technique of machine learning, our method will discover the association rules about pornographic and medical web pages from training data (known web pages), thus filtering the unknown web pages on the basis of these rules. As illustrated in Figure 1, the structure of the proposed method is comprised of three phases: 1) Training Phase, 2) Classification Phase, and 3) Relearning Phase, which will be introduced as follows.

3.1 The Training Phase

The purpose of this phase is to find association rules of differentiating between pornographic, medical, and normal websites by analyzing training web pages. Then, these association rules will be applied to classify the unknown web pages in the Classification Phase.

In this phase, the training web pages should be examined by the Features Extraction Module to extract their critical features (i.e., the values of Critical Attributes) first. Then, the duplicate copies of training web pages

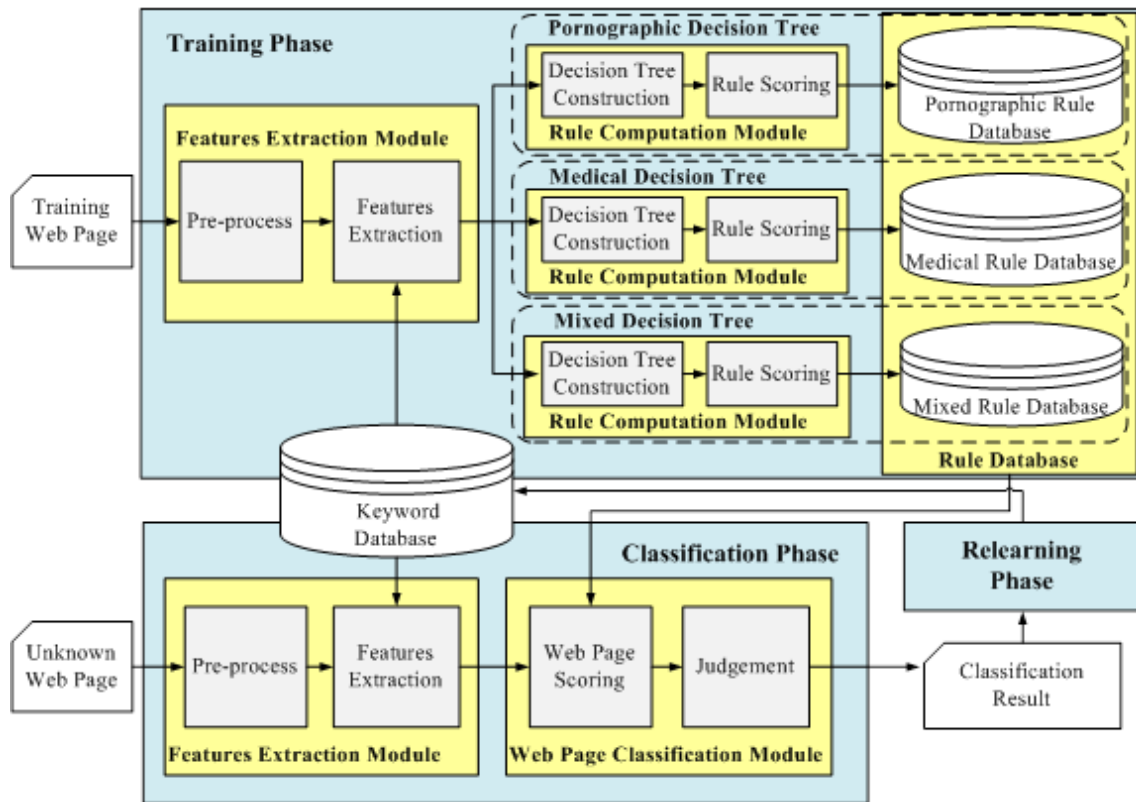


Figure 1: Structure of the proposed method

will be applied simultaneously to construct various decision trees. Note that there are three categories of training web pages: pornographic, medical, and normal. As shown in Figure 1, we construct three decision trees (Pornographic Decision Tree, Medical Decision Tree, and Mixed Decision Tree) and individually equip each decision tree with one Rule Computation Module. By using the copies of pornographic training web pages and normal training web pages as input data, the Rule Computation Module in Pornographic Decision Tree will compute the association rules of distinguishing pornographic websites from normal websites. And the Rule Computation Module in Medical Decision Tree will use the copies of medical training web pages and normal training web pages as input data to compute the association rules of distinguishing medical websites from normal websites. Moreover, the Rule Computation Module in Mixed Decision Tree will use the copies of medical training web pages and pornographic web pages to compute the association rules of distinguishing between medical websites and pornographic websites. Then the resulted rules will be stored respectively into the corresponding rule databases (Pornographic Rule Database, Medical Rule Database, and Mixed Rule Database).

The detailed processes of Features Extraction Module and Rule Computation Module will be discussed as follows.

3.1.1 Features Extraction Module

In the Features Extraction Module, each web page will be analyzed and its Critical Attributes' values will be extracted by applying the following two steps: 1) Pre-process; 2) Features extraction.

The first step is pre-process. In this step, each web page should first be converted into the HTML format, which will be examined by the second step such that its values of Critical Attributes could be verified in the HTML structures.

The second step, features extraction, is designed to discern the critical features of web pages, based on which the suspicious elements of HTML structures that contain relevant keywords will be analyzed. In order to distinguish medical web pages from pornographic ones, judgments will be made based on the features of the HTML head and body, as well as the frequency of medical or pornographic keywords. We study and outline check elements in Table 1 according to the research of Lee et al. [10], which studied and pointed out the parts of this source code mostly likely to be dominated by pornographic keywords. For the sake of convenience, "XXX" will be used to represent the strings that pornographic materials (keywords) might appear. Note that these elements will serve as the Critical Attributes for the computation of association rules in this paper.

As described in Table 1, all these Critical Attributes of each web page will be valued by 0, 1, and 2 by check-

Table 1: The critical attributes used in this study

Type	Critical Attribute		Judgment condition
	No.	Description	
URL	1	Whether there are keywords in the written in HTML Tag of URL.	The URL (URL://XXX) containing pornographic keywords should be set as 1; the URL containing medical keywords should be set as 2; while the URL containing neither should be set as 0.
The head elements	2	Whether there are keywords in the HTML Tag of title.	The HTML Tag <title>XXX</title> containing pornographic keywords should be set as 1; those containing medical keywords should be set as 2; while those containing neither should be set as 0.
	3	Whether there are keywords in the HTML Tag of link (A).	The HTML Tag <link href="XXX"> containing pornographic keywords should be set as 1; those containing medical keywords should be set as 2; while those containing neither should be set as 0.
	4	Whether there are keywords in the HTML Tag of link (B).	The HTML Tag <link title="XXX"> containing pornographic keywords should be set as 1; those containing medical keywords should be set as 2; while those containing neither should be set as 0.
	5	Whether there are keywords in the HTML Tag of metadata (A).	The HTML Tag <meta name="author" content="XXX"> containing pornographic keywords should be set as 1; those containing medical keywords should be set as 2; while those containing neither should be set as 0.
	6	Whether there are keywords in the HTML Tag of metadata (B).	The HTML Tag <meta name="keyword" content="XXX"> containing pornographic keywords should be set as 1; those containing medical keywords should be set as 2; while those containing neither should be set as 0.
	7	Whether there are keywords in the HTML Tag of metadata (C).	The HTML Tag <meta name="description" content="XXX"> containing pornographic keywords should be set as 1; those containing medical keywords should be set as 2; while those containing neither should be set as 0.
	8	Whether there are keywords in the HTML Tag of metadata (D).	Both the HTML Tags <meta name="keyword" content="XXX"> and <meta name="description" content="XXX"> containing pornographic keywords should be set as 1; those containing medical keywords should be set as 2; while those containing neither should be set as 0.
	The body elements	9	Whether there are keywords in the HTML Tag of hyperlink (A).
10		Whether there are keywords in the HTML Tag of hyperlink (B).	The HTML Tag <a>XXX containing pornographic keywords should be set as 1; those containing medical keywords should be set as 2; while those containing neither should be set as 0.
11		Whether there are keywords in the HTML Tag of image (A).	The HTML Tag containing pornographic keywords should be set as 1; those containing medical keywords should be set as 2; while those containing neither should be set as 0.
12		Whether there are keywords in the HTML Tag of image (B).	The HTML Tag containing pornographic keywords should be set as 1; those containing medical keywords should be set as 2; while those containing neither should be set as 0.
13		Whether there are keywords in the HTML Tag of image (C).	The HTML Tag containing pornographic keywords should be set as 1; those containing medical keywords should be set as 2; while those containing neither should be set as 0.
Frequency of keywords	16	There exist 4 to 6 pornographic keywords in the body elements.	The body containing 4 to 6 pornographic keywords should be set as 1; otherwise, as 0.
	17	There exist more than 7 pornographic keywords in the body elements.	The body containing more than 7 pornographic keywords should be set as 1; otherwise, as 0.
	18	There exist 2 to 4 medical keywords in the body elements.	The body containing 2 to 4 medical keywords should be set as 2; otherwise, as 0;
	19	There exist more than 5 medical keywords in the body elements.	The body content containing more than 5 medical keywords should be set as 2; otherwise, as 0;

ing whether the corresponding HTML elements meet the setting conditions.

Note that these Critical Attributes will be examined whether they contain pornographic and medical keywords via the Keyword Database. In this research, the pornographic keywords used are collected from the website SafeSquid², and the medical keywords used in this research are collected from the website MedlinePlus [14]. All these pornographic keywords and medical keywords will be stored respectively into Pornographic Keyword Table and Medical Keyword Table of the Keyword Database in advance. Note that the factual category of each training web page is known. If the training web page is pornographic, its Target Attribute should be valued as "P"; if the training web page is medical, its Target Attribute should be valued as "M"; if the training web page is normal, its Target Attribute should be valued as "N". Then, the acquired Critical Attributes and Target Attribute of web pages should be used to build the decision tree in the Decision Tree Construction Module.

3.1.2 Rule Computation Module

As shown in Figure 1, we apply three copies of Rule Computation Module individually to construct three kinds of decision tree and compute their association rules: Pornographic Decision Tree, Medical Decision Tree, and Mixed Decision Tree. This task of the Rule Computation Module contains two steps: 1) Decision tree construction; 2) Rule scoring.

In the first step, Pornographic Decision Tree, Medical Decision Tree, and Mixed Decision Tree will be constructed respectively. The critical characteristics (i.e., Critical Attributes and Target Attribute) of the related training web pages extracted by the Features Extraction Module are set as the input data in each of the three copies of Rule Computation Module. Then ID3 algorithm will be applied to build decision tree and compute the association rule between the Critical Attributes and Target Attribute.

In the second step, we calculate two kinds of score, pornographic score and medical score, for each association rule resulted from the previous step. Each rule will be scored using the formulas based on the values of its degree of support and degree of purity, which are introduced as follows.

Given an association rule R, assume that leaf node of this rule is C, and $Support(C)$, $Purity(C)$, and $Label(C)$ are defined as mentioned earlier. Let $RuleSupport(R)$ be the support degree of rule R with $RuleSupport(R) = Support(C)$. We compute the values of support degree for all rules, and name the maximum one as RS_{MAX} and the minimum one as RS_{MIN} . Let $|C|$ be the number of data instances in the leaf node C. Assume that n_P is the number of data instances concerning the Target Attribute's value is "P" (i.e., pornographic) and n_M is

the number of data instances concerning the Target Attribute's value is "M" (i.e., medical) in C. The following three functions are necessary for designing the scoring formula of rules: $PornDegree(R)$, $MedicalDegree(R)$ and $Weight(R)$.

The function $Weight(R)$ calculates the weighted value of rule R by the following formula:

$$Weight(R) = \frac{RuleSupport(R)}{RS_{MAX} + RS_{MIN}} \times 100\%.$$

The function $PornDegree(R)$ implies rule's "intensity" to classify web pages as pornographic, which is defined as follows:

$$PornDegree(R) = Purity(C) \text{ if } Label(C) = "P"; \text{ and} \\ PornDegree(R) = \left(\frac{n_P}{|C|}\right) \times 100\% \text{ otherwise.}$$

Moreover, the function $MedicalDegree(R)$ implies rule's "intensity" to classify web pages as medical by the following formulas:

$$MedicalDegree(R) = Purity(C) \\ \text{if } Label(C) = "M";$$

$$\text{and } MedicalDegree(R) = \left(\frac{n_M}{|C|}\right) \times 100\% \text{ otherwise.}$$

Finally, we introduce the formulas of computing pornographic score and medical score for rule R respectively: $PornScore(R)$ and $MedicalScore(R)$. These two formulas are composed of $Weight(R)$ and either $PornDegree(R)$ or $MedicalDegree(R)$ in a ratio of 3:10, which are described as follows:

$$PornScore(R) = (1 \times PornDegree(R) \\ + 0.3 \times Weight(R)) \times 100; \\ MedicalScore(R) = (1 \times MedicalDegree(R) \\ + 0.3 \times Weight(R)) \times 100.$$

By applying the formulas mentioned above, pornographic score and medical score of all rules can be acquired. Then, all rules of three decision trees will be stored into the corresponding rule database, which will be accessed by the Classification Phase to classify unknown web pages.

Moreover, now we define the thresholds in judging the unknown web pages as pornographic or medical for each rule database respectively. In the Pornographic Rule Database, we choose each rule R with $PornDegree(R) \geq 80\%$ and set the minimum pornographic score of the chosen rules as $\lambda(PornRD)$, which will be the threshold of the Pornographic Rule Database for judging the unknown web page is either pornographic or normal used in the Classification Phase. Similarly, we pick out each rule R with $MedicalDegree(R) \geq 80\%$ in the Medical Rule Database and set the minimum medical score of the chosen rules as $\lambda(MedicalRD)$, which will be the threshold of the Medical Rule Database for judging the unknown web page is either pornographic or normal used

²<http://www.safesquid.com/>

in the Classification Phase. Finally, each rule R with $PornDegree(R) \geq 80\%$ in the Mixed Rule Database will be picked out and the minimum pornographic score of the chosen rules will be set as $\lambda(MixedRD)$, which will be the threshold of the Mixed Rule Database for judging the unknown web page is either pornographic or medical used in the Classification Phase.

3.2 The Classification Phase

The purpose of this phase is to examine unknown web pages and classify them as pornographic, medical, or normal. As shown in Figure 1, this phase is comprised of the following two modules: 1) Features Extraction Module and 2) Web Page Classification Module. Firstly, each unknown web page will be inspected by the Features Extraction Module to extract its critical features. Then, the extracted features of this unknown web page will be transmitted to Web Page Classification Module in order to judge its category (pornographic, medical, or normal). The detailed processes of the two modules are described as follows.

3.2.1 Features Extraction Module

The task of Features Extraction Module is basically the same as that of Training Phase. Each unknown web page will be processed by the following two steps: 1) Pre-process; 2) Features extraction. In the first step, each unknown web page will be converted into the HTML format. Then, the second step is to extract the critical features by examining the HTML structure of each unknown web page. By checking the elements outlined in Table 1, the values of 19 Critical Attributes of each unknown web page now can be obtained, which will be used later by the Web Page Classification Module to judge the category of this unknown web page.

3.2.2 Web Page Classification Module

By applying the 19 Critical Attributes extracted in previous module, this Web Page Classification will access the rule databases (Pornographic Rule Database, Medical Rule Database, and Mixed Rule Database) to classify the unknown web pages as pornographic, medical, or normal.

The major steps of algorithm for classifying each unknown web page are as follows:

Step 1. Access the Pornographic Rule Database. This unknown web page will dovetail with some association rule (say, R_1) according to its extracted values of Critical Attributes.

Step 2. Access the Medical Rule Database. Similarly, this unknown web page will dovetail with some association rule (say, R_2) according to the extracted values of Critical Attributes.

Step 3. If $PornDegree(R_1) < \lambda(PornRD)$ and $MedicalDegree(R_2) < \lambda(MedicalRD)$, then this unknown web page will be classified as normal, and stop; else if $PornDegree(R_1) \geq \lambda(PornRD)$ and $MedicalDegree(R_2) < \lambda(MedicalRD)$, then this unknown web page is classified as pornographic, and stop; else if $PornDegree(R_1) < \lambda(PornRD)$ and $MedicalDegree(R_2) \geq \lambda(MedicalRD)$, then this unknown web page is classified as medical, and stop; else if $PornDegree(R_1) \geq \lambda(PornRD)$ and $MedicalDegree(R_2) \geq \lambda(MedicalRD)$, then perform the next step.

Step 4. Access the Mixed Rule Database, and this unknown web page will dovetail with some association rule (say, R_3) according to its extracted values of Critical Attributes. If $PornDegree(R_3) \geq \lambda(MixedRD)$, then this unknown web page will be classified as pornographic; else classify this unknown web page as medical.

3.3 The Relearning Phase

By applying the technique of supervised learning, the task of Relearning Phase is to learn new pornographic or medical keywords incrementally into the Keyword Database. After an unknown web page is judged by Classification Phase, the Relearning Phase will inspect the classification result artificially. In this study, the supervisor will check whether the unknown web page is misjudged. If any misjudgment is produced, the titles and content of the misjudged web pages will then be analyzed and compared to the existing Keyword Database, in order to see whether there are new pornographic keywords or medical keywords. If that is the case, the new keywords will be stored into the Keyword Database.

4 Experimental Design and Results

In this section, we designed and performed experiments to confirm the accuracy and efficiency of the proposed method. In this study, the non-pornographic web pages are composed of medical web pages and normal web pages. In order to measure the performance of this experiment, this study used the decision confusion matrix in Table 2 to estimate the classification results [2]. The purpose of our filtering method is to classify pornographic web pages correctly.

In this research, TP (true positive) means the amount of pornographic web pages that are classified correctly as pornographic; TN (true negative) means the amount of non-pornographic web pages that are classified as non-pornographic web pages. FN and FP refers to misjudgments: FN (false negative) means the amount of pornographic web pages that are misjudged as non-pornographic and FP (false positive) means the amount of

non-pornographic web pages that are misjudged as pornographic.

Table 2: Four cases of judgement

Classification	In reality	
	Pornographic web pages	Non-pornographic web pages
Pornographic web pages	TP (true positive)	FP (false positive)
Non-pornographic web pages	FN (false negative)	TN (true negative)

In this research, the rates of four values of TP, FP, FN, and TN will be computed respectively by the following formulas: $TPR = TP/(TP + FN)$, $FPR = FP/(FP + TN)$, $FNR = FN/(TP + FN)$, and $TNR = TN/(FP + TN)$. Moreover, the three efficacy assessment indexes of "Accuracy", "Precision" and "Recall" will be used to evaluate the filtering accuracy concerning pornographic web pages [23, 25]. Accuracy is used to evaluate the accuracy of the classification results, namely, the proportion of the web pages that are accurately classified to their own categories. It is calculated through the following formula:

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

Precision is used to evaluate the proportion of pornographic web pages among all the web pages that are judged to be pornographic in nature. It is calculated through the following formula:

$$Precision = \frac{TP}{(TP + FP)}$$

Recall is used to evaluate the proportion of the pornographic web pages that are accurately classified as pornographic, which is calculated through the following formula:

$$Recall = \frac{TP}{(TP + FN)}$$

In this research, "F-measure", which is the harmonic mean of precision and recall, is adopted as one of the measuring indexes of the filtering mechanism. It is calculated through the following formula:

$$Fmeasure = \frac{2 \times Precision \times Recall}{Precision + Recall}$$

For example, when the value of precision is too high while the value of recall is too low, it means that although the chances of non-pornographic websites being misjudged are low, the pornographic ones could not be

filtered accurately. Under this circumstance, the value of F-measure would be relatively low, thus signifying the poor filtering effects of this method. F-measure is thus a means of evaluation that could combine precision and recall effectively.

The pornographic web pages, medical web pages and normal pages used in this research were compiled from the website urlblackist.com [25]. This website collected all kinds of web pages from various free websites and updates in a continuous manner. This research eliminated inaccurate web pages, web pages without any content, and web pages whose information is not sufficient. Then, we gathered 2250 web pages for the experiments in this study, including 750 pornographic web pages, 750 medical web pages and 750 normal web pages. The numbers of these web pages used in the Training Phase and the Classification Phase of the proposed filtering method were shown in Table 3. Note that the training and unknown web pages should be selected randomly from the three categories.

In the Training Phase, 900 web pages were selected randomly according to the ratio 1:1:1 and trained as three combinations. The training task was performed by three decision trees: Pornographic Decision Tree, Medical Decision Tree, and Mixed Decision Tree. The Pornographic Decision Tree contained 300 pornographic web pages and 300 normal web pages, the Medical Decision Tree contained 300 medical web pages and 300 normal web pages, while the Mixed Decision Tree was the mixture of 300 medical web pages and 300 pornographic web pages.

To confirm the accuracy and efficiency of the proposed filtering method, we performed three experiments, which examined the following performances: (A) the effectiveness of the proposed method in avoiding the misjudgment of medical web pages, (B) the effectiveness of the Relearning Phase, and (C) the stability of the proposed method. These experimental results will be discussed as follows.

(A) The effectiveness of the proposed method in avoiding the misjudgment of medical web pages

The purpose of this experiment was to confirm the effectiveness of the proposed method in avoiding the misjudgment of classifying medical web pages as pornographic ones. In this experiment, we chose randomly 300 medical, 300 pornographic, and 300 normal web pages as the unknown web pages, which would be inputted into the Classification Phase.

Firstly, we performed the Classification Phase without using the Medical Keyword Table (i.e., let the Medical Keyword Table be empty). As shown in Table 4, the number of misjudged medical web pages was 71, while the number of misjudged normal ones was 4. Obviously, the misjudgments of medical web pages were more frequent than that of normal web pages if we omitted the Medical Keyword Table. According to the filtering results of Table 5, the proportion of pornographic web pages that were accurately filtered was (TPR) 97.33%, while the proportion of non-pornographic web pages that were accurately

Table 3: The numbers of web pages used in each phase

	Training web pages in the Training Phase	Unknown web pages in the Classification Phase	Total Total
Pornographic web pages	300	450	750
Medical web pages	300	450	750
Normal web pages	300	450	750
Total	900	1350	2250

Table 4: The classification result of medical web pages and normal ones

	Medical web pages	Normal web pages
The number of misjudged web pages	71	4
The number of web pages judged correctly	239	296
Total	300	300

filtered was (TNR) 87.52%.

Table 5: The efficiency of classification without using the Medical Keyword Table

Indexes	Measurement	Indexes	Measurement
TPR	97.33%	Accuracy	90.79%
TNR	87.52%	Recall	97.04%
FPR	2.67%	Precision	87.52%
FNR	12.48%	F-measure	83.01%

Then, we perform the Classification Phase by applying the Medical Keyword Table. As shown in Table 6, the number of misjudged medical web pages was reduced obviously. This means that after the designed application of the Medical Keyword Table, the filtering accuracy of our method was improved. Moreover, Table 7 recorded the classification efficiency of this experiment. By comparing Table 7 with Table 5, we observed that all the efficacy assessment indexes of Accuracy, Precision, Recall, and F-measure were improved noticeably. Moreover, FPR decreases from 12.48% (before the Medical Keyword Table was used) to 3.67%. Thus, we can deduce that the systematic method proposed in this study will effectively reduce the misjudgments of classifying non-pornographic websites as pornographic ones.

(B) The effectiveness of the Relearning Phase

The purpose of this experiment was to examine the effectiveness of the Relearning Phase of the proposed method. In this experiment, we used 450 medical, 450 pornographic, and 450 normal web pages as the unknown web pages, which would be inputted into the Classification Phase.

The experimental results were shown in Table 8 and 9. The case (I) meant that the Relearning Phase was

turned off, and the case (II) indicated that the Relearning Phase was turned on during the classification of unknown web pages. As shown in Table 8, the numbers of misjudged web pages of case (II) were all less than that of case (I), which implied that the Relearning Phase could effectively decrease the probability of misjudgment. Moreover, the values of efficacy assessment indexes were recorded in Table 9. By using the Relearning Phase, the evaluation indicator FPR (the rate of non-pornographic web pages being misjudged as pornographic) decreased from 4.68% to 1.64%. Moreover, the Accuracy increased from 96.21% to 98.26% while the Precision increased from 95.32% to 98.36%. This means that both the Accuracy and Precision were improved after the Relearning Phase was turned on; after the re-learning, TPR (the rate of pornographic web pages being accurately judged as pornographic) increased from 97.95% to 97.99% while FNR (the rate of pornographic web pages being judged as non-pornographic) decreased from 2.05% to 2.01%, showing a slight improvement in terms of the filtering performance concerning pornographic web pages. TNR (the rate of non-pornographic web pages classified accurately as non-pornographic) increased from 95.32% to 98.36%, a significant increase in terms of the classification of normal web pages. FPR decreased from 4.68% to 1.64%, a substantial improvement in terms of the misjudgment rate. These results showed that the relearning mechanism would improve the classification capabilities and performance of the proposed filtering method in this paper.

(C) Testing of the stability

This experiment was set out to investigate whether the classification performance of our method proposed in this paper will be influenced when the data was combined in a different ratio. While the original ratio between normal, pornographic and medical web pages was 1:1:1, some tests were conducted in this experiment over the three kinds of web pages under various ratios, with the aim to guarantee

Table 6: The improved classification result of medical web pages and normal ones

	Medical web pages	Normal web pages
The number of misjudged web pages	18	4
The number of web pages judged correctly	282	296
Total	300	300

Table 7: The classification efficiency of using the Medical Keyword Table

Indexes	Measurement	Indexes	Measurement
TPR	97.33%	Accuracy	96.67%
TNR	96.33%	Recall	97.31%
FPR	2.67%	Precision	96.34%
FNR	3.67%	F-measure	95.86%

Table 8: The number of misjudged web pages

	Medical web pages		Normal web pages		Pornographic web pages	
	Case (I)	Case (II)	Case (I)	Case (II)	Case (I)	Case (II)
The number of misjudged web pages	21	11	8	4	11	7
The number of web pages judged correctly	429	439	442	446	439	443
Total	450	450	450	450	450	450

Table 9: The effectiveness of the relearning phase

Indexes	Measurement		Indexes	Measurement	
	Case (I)	Case (II)		Case (I)	Case (II)
TPR	97.95%	97.99%	Accuracy	96.21%	98.26%
TNR	95.32%	98.36%	Recall	97.89%	98.00%
FNR	2.05%	2.01%	Precision	95.32%	98.36%
FPR	4.68%	1.64%	F-measure	94.06%	98.54%

Table 10: The experimental results of six data groups under various combination ratios

Group No.	Total number of web pages	Ratio	Accuracy (%)	Precision (%)	FNR (%)	FPR (%)
1	900	1:1:2	98.07	97.93	1.78	2.07
		1:2:1	98.22	98.01	1.56	2.00
		2:1:1	98.22	98.65	2.22	1.33
2	900	1:1:3	98.17	97.69	1.33	2.33
		1:3:1	98.06	97.68	1.56	2.33
		3:1:1	98.33	98.66	2.00	1.33
3	900	1:1:5	98.43	97.98	1.11	2.04
		1:5:1	98.39	98.34	1.56	1.67
		5:1:1	98.24	98.69	2.22	1.30
4	600	1:1:2	98.33	98.67	2.00	1.33
		1:2:1	98.33	98.33	1.67	1.67
		2:1:1	98.11	98.22	2.00	1.78
5	600	1:1:3	97.88	97.76	2.00	2.25
		1:3:1	98.33	98.01	1.33	2.00
		3:1:1	98.13	98.25	2.00	1.75
6	600	1:1:5	98.19	98.06	1.67	1.94
		1:5:1	98.50	98.34	1.33	1.67
		5:1:1	98.61	98.88	1.67	1.11

the stability of the filtering mechanism adopted in the current research. Table 10 shows the experimental results of data groups under the different classifications. Note that three tests were conducted for each group, and the three kinds of web pages of each group were combined according to the designated ratio. We give an example as follows. Let the total number of web pages in a certain group be 600 and the ratio designated for some test be 1:2:3. Therefore, the web pages in this test will be composed of 100 normal, 200 pornographic, and 300 medical web pages.

Obviously, some changes occurred over the four measuring indicators of Accuracy, Precision, FNR and FPR, though not very substantial changes; when medical web pages accounted for a higher proportion, the FPR (the proportion of non-pornographic pages being misjudged as pornographic) in most groups decreased slightly, but not so much different from the value when the ratio was 1:1:1. This indicated that the filtering results of our method would not be greatly influenced by the changes in the data. In terms of the misjudgment of medical web pages, the values of precision and FPR were fair proof that the method in this research was satisfactory.

5 Conclusions

Concerning the past filtering mechanisms of pornographic web pages, the difficulties in distinguishing medical web pages from pornographic ones have baffled the users of medical websites for a long time. The filtering method proposed in this paper works by selecting the features of the web pages and establishing decision trees according to the category of web pages. Then, the resulted association rules in each decision tree are applied to filter the unknown web pages. To confirm the accuracy and efficiency of the proposed filtering method, we performed three experiments. The first experiment was to examine the effectiveness of the proposed method in avoiding the misjudgment of medical web pages. According to the decrease of FPR, we could deduce that the systematic method proposed in this study would effectively reduce the misjudgments of classifying non-pornographic websites as pornographic ones. The second experiment was to examine the effectiveness of the Relearning Phase. The results showed that the relearning mechanism improved the classification capabilities and performance of the proposed filtering method conspicuously. The experimental results of the third experiment indicated that the filtering results of our method would not be greatly influenced by the changes in the data composition. The Accuracy of this research reached a satisfactory value (greater than 98%). Moreover, the value of F-measure was 98.54%, which showed that the values of Precision and Recall also reached the satisfactory standards, without any figure that's extremely high or extremely low. Therefore, we can conclude that the filtering method proposed in this research is satisfactory because of its outstanding performance and effectivity.

Acknowledgments

This work is supported by the Ministry of Science and Technology, Taiwan, R.O.C. under Grant no. MOST 103-2410-H-004-112.

References

- [1] A. Ahmadi, M. Fotouhi, and M. Khaleghi, "Intelligent classification of web pages using contextual and visual features," *Applied Soft Computing*, vol. 11, no. 2, pp. 1638–1647, 2011.
- [2] I. Androutsopoulos, J. Koutsias, K. V. Chandrinos, G. Paliouras, and C. D. Spyropoulos, "An evaluation of naive bayesian anti-spam filtering," in *11th European Conference on Machine Learning*, pp. 9–17, 2000.
- [3] M. M. Fleck, D. A. Forsyth, and C. Bregler, "Finding naked people," in *Computer Vision (ECCV'96)*, pp. 593–602, 1996.
- [4] M. Hammami, Y. Chahir, and L. Chen, "Webguard: A web filtering engine combining textual, structural, and visual content-based analysis," *IEEE Transactions on Knowledge and Data Engineering*, 1vol. 8, no. 2, pp. 272–284, 2006.
- [5] W. H. Ho, and P. Watters, "Statistical and structural approaches to filtering internet pornography," in *IEEE International Conference on Systems, Man and Cybernetics*, vol. 5, pp. 4792–4798, 2004.
- [6] ICRA, *Internet Content Rating Association*, Mar. 29, 2017. (<http://www.fosi.org/icra/>)
- [7] T. Kajiyama, and I. Echizen, "An educational system to help students assess website features and identify high-risk websites," *Interactive Technology and Smart Education*, vol. 12, no.1, pp. 14–30, 2015.
- [8] M. Kanuga, and W. D. Rosenfeld, "Adolescent sexuality and the internet: the good, the bad, and the URL," *Journal of Pediatric and Adolescent Gynecology*, vol. 17, no. 2, pp. 117–124, 2004.
- [9] L. H. Lee, and C. J. Luh, "Generation of pornographic blacklist and its incremental update using an inverse chi-square based method," *Information Processing & Management*, vol. 44, no. 5, pp. 1698–1706, 2008.
- [10] P. Y. Lee, S. C. Hui, and A. C. M. Fong, "An intelligent categorization engine for bilingual web content filtering," *IEEE Transactions on Multimedia*, vol. 7, no. 6, pp. 1183–1190, 2005.
- [11] D. Li, N. Li, J. Wang, and T. Zhu, "Pornographic images recognition based on spatial pyramid partition and multi-instance ensemble learning," *Knowledge-Based Systems*, vol. 84, pp. 214–223, 2015.
- [12] T. M. Mahmoud, T. Abd-El-Hafeez, and A. Omar, "An Efficient System for Blocking Pornography Websites," in *Computer Vision and Image Processing in Intelligent Systems and Multimedia Technologies*, IGI Global, pp. 161–176, 2014.

- [13] J. A. Marcial-Basilio, G. Aguilar-Torres, G. Sanchez-Perez, L. K. Toscano-Medina, and H. M. Perez-Meana, "Detection of pornographic digital images," *International Journal of Computers*, vol. 5, no. 2, pp. 298–305, 2011.
- [14] MedlinePlus, <http://www.nlm.nih.gov/medlineplus/>.
- [15] M. G. Noll, and C. Meinel, "Web page classification: An exploratory study of the usage of Internet content rating systems," in *LIASIT-Luxembourg International Advanced Studies in Information Technologies*, Luxembourg, 2005.
- [16] C. Ohmann, V. Moustakis, Q. Yang, K. Lang, and Acute Abdominal Pain Study Group, "Evaluation of automatic knowledge acquisition techniques in the diagnosis of acute abdominal pain," *Artificial Intelligence in Medicine*, vol. 8, no. 1, pp. 23–36, 1996.
- [17] Pew Internet, *What the Public Knows About Cybersecurity*, Mar. 22, 2017. (<http://pewinternet.org/>)
- [18] PICS, *Platform for Internet Content Selection*, Mar. 29, 2017. (<http://www.w3.org/PICS>)
- [19] J. R. Quinlan, "Induction of decision trees," *Machine learning*, vol. 1, no. 1, pp. 81–106, 1986.
- [20] J. R. Quinlan, *C4.5: Programs for Machine Learning*, Elsevier, 2014.
- [21] B. H. Schell and C. Martin, *Cybercrime: A Reference Handbook*, ABC-CLIO, 2004.
- [22] K. D. Stark and D. U. Pfeiffer, "The application of non-parametric techniques to solve classification problems in complex data sets in veterinary epidemiology-an example," *Intelligent Data Analysis*, vol. 3, no. 1, pp. 23–35, 1999.
- [23] G. Y. Su, J. H. Li, Y. H. Ma, and S. H. Li, "Improving the precision of the keyword-matching pornographic text filtering method using a hybrid model," *Journal of Zhejiang University Science*, vol. 5, no. 9, pp. 1106–1113, 2004.
- [24] L. Sui, J. Zhang, L. Zhuo, and Y. C. Yang, "Research on pornographic images recognition method based on visual words in a compressed domain," *IET Image Processing*, vol. 6, no. 1, pp. 87–93, 2012.
- [25] URL blacklist service, Mar. 29, 2017. (<http://urlblacklist.com/>)
- [26] J. Zhang, L. Sui, L. Zhuo, Z. Li, and Y. Yang, "An approach of bag-of-words based on visual attention model for pornographic images recognition in compressed domain," *Neurocomputing*, vol. 110, pp. 145–152, 2013.
- [27] L. Zhuo, J. Zhang, Y. Zhao, and S. Zhao, "Compressed domain based pornographic image recognition using multi-cost sensitive decision trees," *Signal Processing*, vol. 93, No. 8, pp. 2126–2139, 2013.

Biography

Jyh-Jian Sheu is currently an associate professor in College of Communication, National Chengchi University, Taiwan. He received his B.B.A. degree in Management Information Systems from National Chengchi University, Taiwan, and his M.S. and Ph.D. degrees in Computer and Information Science from National Chiao Tung University, Taiwan. His primary research interests include data mining, Internet security, and Big Data.

Privacy-preserving Similarity Sorting in Multi-party Model

Yifei Yao¹ and Fanhua Yu¹

(Corresponding author: Yifei Yao)

College of Computer Science and Technology, Changchun Normal University¹

No. 677, Changji North Road, Erdao District, Changchun 130032, China

(Email: yao-yifei@126.com)

(Received May. 29, 2016; revised and accepted Aug. 25 & Sep. 3, 2016)

Abstract

In social network, it is conceivable that a rational execution sequence does good to cooperative mission, especially for a large number of participants. However, there are many difficulties for multi-party computation, the most important of which is privacy. In this paper, secure multi-party computation technology and dimensionality reduction are chosen to design a privacy-preserving protocol, which sorts m people according to their similarity. In a n dimensional system, the secure protocol's time complexity is $O(mn + n + m \log m)$ and communication complexity is $O(m)$. Detailed analysis about security and applicability are also presented in this paper. In addition, the protocol can be improved in security at the cost of complexity, with an arbitration agreement designed against fraud.

Keywords: Dimensionality Reduction; Privacy-preserving Computation; Secure Multi-party Computation; Similarity Sort Algorithm

1 Introduction

In the age of big data, complex information is emerging endlessly, and traditional algorithms are facing challenges of high dimensional data. Meanwhile, disclosure of sensitive information becomes the major deterrent for the growth of social network [1]. For these issues, special schemes have been proposed in many domains, such as designing privacy-aware systems in a cloud environment [9] and defining privacy protection mechanisms for mobile social networks [11, 12]. In contrast to developing approaches against corruption attacks, arbitral protocols is also a good choice for fairness and privacy preserving.

With the rapid development of communication technology, security turns more and more essential, which makes secure protocols designed to solve basic problem popular [14]. In former applications, people always collect distributed information together and turn to a trust third party (TTP) for solution. But the demand of pri-

vacancy makes it hard to find such an agency trusted by all the participants. Actually, each party wants the result correct, avoiding leaking his information to the others. Secure multi-party computation makes cooperative calculation privately and prevents participants' data from leaking [5]. Privacy-preserving techniques provide methods to calculate functions with the input of private information [18]. It turns out to be attractive because it can benefit people in security [4].

One significant technical challenge in social network is sorting. With an execution sequence for the participants, they will work more efficiently and fairly. In addition, sorting algorithm is the basis of many fields such as data analysis and database systems, which is a core step of many algorithms and of significance both in theory and practice. Guan Wang pushes all knowledge and influence of input values down to small black-box circuits avoiding the significant computational overheads, he uses Yao's garbled paradigm as reference, but his method only works on two party system. Doctor Jónsson proposed a secure sort algorithm which can be used as a building-block with $O(n \log n)$ comparisons in $O(\log n)$ rounds. Even it can be built upon any secret sharing scheme supporting multiplication and addition, complexity is high without any dimensionality reduction [8]. Because it is not easy to construct the optimal branching program for a complex function, Bingsheng Zhang designed several constant-round 0-error oblivious sorting algorithms together with some useful applications. In paper [2], Dan Bogdanov and his partners compared several published sorting methods. They evaluated the theoretical performance and discussed the practical implications of the different approaches. After that, Dan Bogdanov's group improved two earlier designs based on sorting networks and quick sort with the capability of sorting matrices. They also proposed two new designs - a naive comparison-based sort with a low round count and an oblivious radix sort algorithm that does not require any private comparisons in [3]. Koki Hamada proposed a simple and general approach of converting non-data-oblivious comparison sort algorithm. Although his

method improved the running time compared to existing protocols in experiments, it can be only used in a certain field as the author described [7]. Then in 2014, Koki Hamada improved his work, he used a new technique called "shuffle-and-reveal" for an $O(n \log n)$ communication complexity result. But it is also restricted for a constant number of parties and a field with a constant size [6].

In this paper, we propose a protocol to achieve a reasonable sequence for m partners in n dimensional system, and then analyze its security, complexity and applicability. The paper is organized as follows. In Section 2 we describe preliminaries. The privacy preserving similarity sorting protocol is introduced in Section 3. Then Section 4 discusses the protocol's complexity, security and applicability. Two kinds of improvement measure are applied in Section 5 and arbitration procedure which is used in case of fraud is discussed in Section 6. The further work together with a conclusion is proposed at last.

2 Preliminaries

2.1 Secure Multi-party Computation (SMC)

SMC is a kind of distributed calculation, it needs each party's private data as input, then broadcasts the final result without leaking anyone's privacy. SMC technology was proposed in 1982 [15], and it comes into more and more domains such as social networking services [17], ad hoc networks [13], computation geometry [16], data mining [10] and so on.

A third-party who is trusted by the whole group can help them do the privacy-preserving work, he can get enough information to complete the calculation and publish the result. But the hypothesis of trusted third party is unsafe and less realistic. It is known that any secure computation problem can be solved by a circuit protocol, but the size of the corresponding circuit is usually too large to realize [15]. So researchers choose to design special protocol for special use in a viable way.

2.2 Secure Sum Protocol

Suppose there are $m \geq 3$ parties P_1, P_2, \dots, P_m who join in the computation. Each participant $P_j (j = 1, 2, \dots, m)$ has his private information d_j . They want to calculate the function $\sum_{j=1}^m d_j$ together, but no one will leak his secret to others. Secure sum protocol solve the problem with the help of data disrupt technique in [15], and Figure 1 shows the meaning of this technique.

- 1) First of all, $P_j (j = 1, 2, \dots, m)$ generates m random shares $x_{j,k}$ for $k = 1, 2, \dots, m$, such that $d_j = \sum_{k=1}^m x_{j,k}$;
- 2) Then, $P_j (j = 1, 2, \dots, m)$ sends P_k with $x_{j,k}$, for $k = 1, 2, \dots, m$ and $k \neq j$;

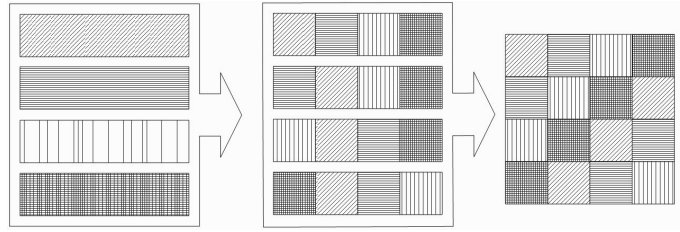


Figure 1: Schematic diagram of secure sum protocol

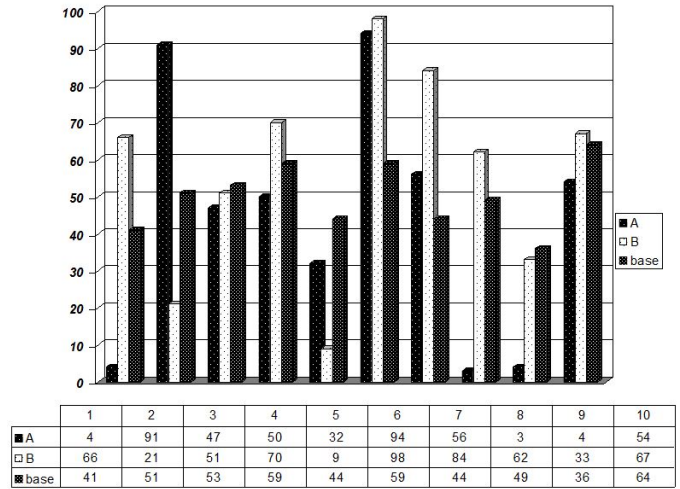


Figure 2: Case of similarity comparison

- 3) After $P_j (j = 1, 2, \dots, m)$ gets all the $x_{k,j}$ from P_k that $k = 1, 2, \dots, m$ and $k \neq j$, he computes $\hat{x}_j = \sum_{k=1}^m x_{k,j}$ and broadcasts it;
- 4) At last, $P_j (j = 1, 2, \dots, m)$ computes $X = \sum_{j=1}^m \hat{x}_j$.

2.3 Similarity Comparison Algorithm

In this paper, similarity comparison algorithm is chosen for dimensionality reduction. For the n dimension system, A and B want to know who is more similar as the baseline $Base$. Suppose $D_A = (a_1, a_2, \dots, a_n)$, $D_B = (b_1, b_2, \dots, b_n)$ and $D_{Base} = (e_1, e_2, \dots, e_n)$, similarity comparison algorithm compare $c_A = \sum_{k=1}^n a_k \times e_k$ and $c_B = \sum_{k=1}^n b_k \times e_k$.

Take numbers in Figure 2 as example, the similarity comparison algorithm says A is closer to $Base$ than B for $c_A = 23411$ and $c_B = 28998$.

2.4 Secret Comparison Protocol

In 1982, A.C. Yao brought forward the famous millionaires problem in [15]: two millionaires, Alice and Bob, want to know which is richer, without revealing their respective wealth. In 2004, Qin brings forward a method for two parties comparing if $a = b$ privately. The method validates its security by computational indistinguishabil-

ity through homomorphism encryption and Φ -hiding assumption. It returns which one is greater or equal to the other, while Yao’s method couldn’t return the equality message. This protocol is complex in computation and safe to resist decoding, it reduces the communication of random data perturbation techniques in Yao’s method.

In 2009, Doctor Luo proposed a three-round protocol for solving the secure comparison problem in the semi-honest setting based on the property of the cross products, who improved cross products protocol by using the Paillier’s additive homomorphic encryption firstly. His method can compare two real numbers in addition to integers, and determine whether $a > b$, $a < b$ or $a = b$ for two partners.

Secret comparison protocol is one of the most important protocols in SMC, it was widely used in privacy-preserving computation geometry [16], social networking services [17] and so on.

Equality-testing is a special kind of secure comparison protocol. It helps two parties know whether their private data are equal or not, moreover, nobody can seek the other’s information from the result if they are not the same.

2.5 Secure Scalar Product Protocol

Secure scalar product protocol is a basic tool in SMC, and it has been applied to the privacy-preserving cooperative calculation widely. It can help two partners compute the scalar product of their private vectors correctly and securely. Suppose Alice has a private vector $X = (x_1, x_2, \dots, x_n)$ and Bob has his own $Y = (y_1, y_2, \dots, y_n)$. After the protocol, Alice get $u = X \cdot Y + v = \sum_{i=1}^n x_i \times y_i + v$ where v is a random number selected by Bob. Meanwhile Alice cannot get any information about $X \cdot Y$ or y_i from u , Bob can get nothing about u or x_i from v either.

3 Privacy-preserving Similarity Sorting Protocol

3.1 Computational Model

Generally speaking, there are potential malicious attacks against any multi-party protocol. In this paper, we study the problem under a semi-honest model, in which each party follows the protocol without trying to intermit or disturb with dummy data, even they will keep a record of all its intermediate computation [15]. This model is practical and useful, because everybody in the cooperation expects the right result rather than others’ private information.

3.2 Security Model

The classical definition of security is stated in [4]. Let f be a function that $P_j(j = 1, 2, \dots, m)$ will compute cooperatively. If there is a protocol Π , for each P_j it can generate

a simulator which can get all messages though the process only with its view and output, then it is secure. It is to say: The protocol Π to compute function f is secure when it satisfies the conditions as follows: There exists a probabilistic polynomial-time simulator $S_j(j = 1, 2, \dots, m)$, it holds that

$$\{(S_j(x_j, t_j), t_1, t_2, \dots, t_{j-1}, t_{j+1}, \dots, t_m)\} \equiv \{view_j^\Pi(x_1, \dots, x_m), v_1, v_2, \dots, v_{j-1}, v_{j+1}, \dots, v_m\} \quad (1)$$

where

$$t_j = f_j(x_1, x_2, \dots, x_m)$$

and

$$v_j = output_j^\Pi(x_1, x_2, \dots, x_m)$$

While the party’s view consists of its initial input, an auxiliary initial input (which is relevant only for modeling adversarial strategies), its random-tape, and the sequence of message it has received so far.

3.3 Privacy-preserving Similarity Sorting Protocol

Input: There are m members in this group, each one has his private data in the form of $D_j = (d_{1,j}, d_{2,j}, \dots, d_{n,j})^T$ which $j = 1, 2, \dots, m$.

Output: Reasonable sorting consequence for the group.

Algorithm 1 Privacy-preserving similarity sorting protocol

- 1: Begin
- 2: Set the quantitative standard.
Everybody agrees to take part in the n -dimensional coordinate system after the preprocess stage.
- 3: all the participants join in the secure sum protocol for n times to get the sum s_i for each dimension, that

$$s_i = d_{i,1} + d_{i,2} + \dots + d_{i,m}$$

Then each one gets the average

$$\bar{E} = (\frac{s_1}{m}, \frac{s_2}{m}, \dots, \frac{s_n}{m})^T = (e_1, e_2, \dots, e_n)^T.$$

- 4: each participant $P_j(j = 1, 2, \dots, m)$ calculates the cross product $c_j = \sum_{k=1}^n d_{k,j} \times e_k$ with his own D_j and the public average \bar{E} .
 - 5: P_j broadcasts his c_j and gains the order.
 - 6: End
-

4 Analysis

In this section, we analyze the complexity, security and applicability for this protocol.

4.1 Complexity Analysis

Conclusion 1: The time complexity of privacy-preserving similarity sorting protocol is $O(mn + n +$

$m \log m$), while there are m partners in n dimensions system.

In a distributed algorithm without leader, time complexity means the time each party spending on the work locally.

In Step 3, secure sum protocol costs each member $m-1$ times random number generation and $2m-2$ times addition. In total, it is $O(mn)$ times addition. In Step 4, each party needs n times multiplication and $n-1$ times addition. In Step 5, there is a sorting algorithm finished in $O(m \log m)$.

In addition, the time complexity turns to be $O(m \log m)$ when $m \gg n$, and be $O(mn)$ when $m \ll n$.

Conclusion 2: The communication complexity of privacy-preserving similarity sorting protocol is $O(m)$ that m is partner number in the group.

There is no interactive communication in Step 4 and Step 5. Only Step 3 sends and receives one piece of message towards each other. In total, the protocol spends $2(m-1)$ messages which is $O(m)$ in short.

4.2 Security Analysis

Conclusion 3: Privacy-preserving similarity sorting protocol can execute securely without leaking privacy.

Now, we analyze the message leaked at each step in this protocol.

- *Secure sum protocol (Step 3):* Each member $P_j (j = 1, 2, \dots, m)$ taking part in the secure sum protocol will get a view as below:

$$\begin{aligned} view_j &= \{d_{i,j}, piece_{i,j,k}, piece_{i,k,j}, s_i, e_i, m\} \\ & i = 1, 2, \dots, n; j = 1, 2, \dots, m; j \neq i; \\ & k = 1, 2, \dots, m \end{aligned}$$

where $(d_{1,j}, d_{2,j}, \dots, d_{n,j})^T$ is P_j 's private data. $piece_{i,A,B}$ is the piece of data sent from member A to B on the i th dimension, satisfying $\sum_{j=1, k=1}^{m,m} piece_{i,j,k} = d_{i,j}$ and $\sum_{j=1, k=1}^{m,m} piece_{i,k,j} = s_i$. s_i is the result of secure sum protocol on the i th dimension, and e_i is the average that $e_i = \frac{s_i}{m}$. Meanwhile, m is the number of people in the calculation group.

From the view, no one can analyze other's private data at all. Even $d_{i,j} = \sum_{k=1}^m piece_{i,j,k}$, $\sum_{k=1}^m piece_{i,k,j}$ means nothing. At the end of this step, the sum and average on each dimension is known by everybody, while nothing sensitive is leaking.

- *Cross Product (Step 4):* Everybody computes the cross product locally and no information can be got by others.
- *Broadcast and Sorting Phase (Step 5):* Even P_j knows $c_l = \sum_{k=1}^n d_{k,l} \times e_k$ where $l = 1, 2, \dots, m$, he

can't get anything more about $d_{k,l} (k = 1, 2, \dots, n)$. Because on each dimension, P_j knows only one equation against n unknown numbers, the mathematical analysis helps keeping secret. After knowing c_l , P_j calls sorting algorithm locally for his order at last.

Because the three steps are independence and there is no rule between them, neither can analyze to know the others' information through the privacy-preserving protocol. This does preserve the parties' privacy.

Furthermore, if there are some people dishonest joining hands for other's privacy, privacy-preserving similarity sorting protocol can hold on security at a certain extent.

$$\mathbf{D} = (D_1, D_2, \dots, D_m) = \begin{bmatrix} d_{1,1} & d_{1,2} & \dots & d_{1,m} \\ d_{2,1} & d_{2,2} & \dots & d_{2,m} \\ \dots & \dots & \dots & \dots \\ d_{n,1} & d_{n,2} & \dots & d_{n,m} \end{bmatrix}$$

For example, participants $\tilde{P}_j = \{P_k | k = 1, \dots, m; k \neq j\}$ join together to expose P_j . What they want to know is $D_j = (d_{1,j}, d_{2,j}, \dots, d_{n,j})^T$ and what they know is $\bar{E} = (\frac{s_1}{m}, \frac{s_2}{m}, \dots, \frac{s_n}{m})^T = (e_1, e_2, \dots, e_n)^T$ that $s_i = d_{i,1} + d_{i,2} + \dots + d_{i,m}$ and $c_j = \sum_{k=1}^n d_{k,j} \times e_k$. They can only reveal n unknown numbers from one certain equation, and when they want to use $s_i = d_{i,1} + d_{i,2} + \dots + d_{i,m}$ there will be more unknown numbers appear in their equations. Even more, it turns to be more and more unprocurable when n turns bigger.

Another risk is the secure sum protocol in Step 3, the protocol needs $m \geq 3$ people, while it is secure when there is no more than $m-2$ dishonest co-conspirators.

4.3 Applicability Analysis

- *Case study:* Take $n = 10$ and $m = 10$ for example, 100 random numbers from 1 to 100 are showing in Table 1.

Table 1: Case study for $n = 10$ and $m = 10$

P_1	P_2	P_3	P_4	P_5	P_6	P_7	P_8	P_9	P_{10}
4	66	61	4	47	8	90	85	77	45
91	21	34	97	81	31	61	2	44	57
47	51	46	73	17	27	28	93	76	23
50	70	54	60	51	63	58	35	66	83
32	9	61	37	35	14	68	6	69	99
94	98	2	97	52	86	27	45	63	52
56	84	13	71	38	48	13	64	46	25
3	62	20	69	82	46	37	63	9	80
4	33	92	52	83	15	45	54	20	8
54	67	63	35	39	69	69	88	32	69
3^{rd}	9^{th}	1^{st}	10^{th}	6^{th}	2^{nd}	4^{th}	7^{th}	5^{th}	8^{th}

The average \bar{E} calculated in Step 3 shows at the last column. Then, Step 4 public each one's cross product value as (23405, 29451, 21925, 30595, 26375, 22007,

25199, 27157, 25723, 27960). At last, they get the sort sequence shows in Figure 3.

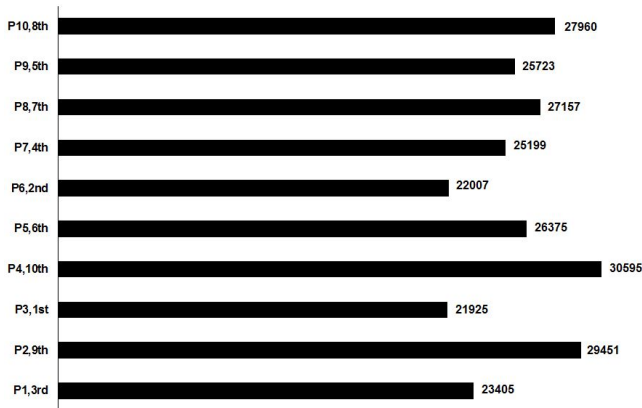


Figure 3: Sequence after sorting

- **Advanced by weight modified:** This protocol can be modified by adding weight w_k to the k^{th} dimension for some reason. In this case, Step 4 calculates $c_j = \sum_{k=1}^n d_{k,j} \times e_k \times w_k$ for a new rank rule. It helps the group keep the flexibility of adjusting weight for a new sort sequence. It is important to note that proposing weight modified protocol many times in the same group must be avoided. Or else, it turns insecure for their privacy.

5 Improved Protocol

In this session, privacy-preserving similarity sorting protocol is improved in security.

5.1 Advanced by Secure Comparison Protocol

In Step 5, P_j broadcasts his c_j and gains the order. If the group doesn't want the sort sequence but only two people's relationship, they can use secure comparison protocol instead. In this way, the two people get the right result without leaking information about c_j . It is a secure method at a higher level taking complexity in exchange.

The improved protocol shows in Algorithm 2:

Input: There are m members in this group, each one has his private data in the form of $D_j = (d_{1,j}, d_{2,j}, \dots, d_{n,j})^T$ which $j = 1, 2, \dots, m$.

Output: The relationship of P_a and P_b .

If there are l people wants the sort sequence while $2 < l < m$, the number of comparison time will be $\log_2 l$ according to the dichotomy.

5.2 Advanced by Data Compression

Data compression can not only strengthen security, but also reduce complexity. After the analysis of test numbers

Algorithm 2 Advanced Protocol

- 1: Begin
- 2: Set the quantitative standard.
- 3: All the participants join in the secure sum protocol for n times to get the sum

$$s_i = d_{i,1} + d_{i,2} + \dots + d_{i,m}$$

for each dimension.

Then each one gets the average

$$\bar{E} = (\frac{s_1}{m}, \frac{s_2}{m}, \dots, \frac{s_n}{m})^T = (e_1, e_2, \dots, e_n)^T.$$

- 4: P_a calculates $c_a = \sum_{k=1}^n d_{k,a} \times e_k$ and P_b calculates $c_b = \sum_{k=1}^n d_{k,b} \times e_k$.
 - 5: Secure comparison protocol helps P_a and P_b get their relationship.
 - 6: End
-

for many times, there are two methods adding data compression technology to the privacy-preserving similarity sorting protocol.

- **Compression before calculation:** In this case, Step 2 of Algorithm 1 should be modified below:

2: In each dimension, the group agrees on the standard that divided in 100 degrees. Each partner's private data can be map in the standard and $d_{i,j}$ should be an integer from 1 to 100.

In this method, privacy is hid by compression before calculation, because $d_{i,j}$ is closely related to partner's private data but not the exact value. By the way, the compression maps values into integers, it simplifies the calculation at the next steps.

This kind of compression also imports some more tasks. If the group wants the preprocess's standard, they must spend time on getting boundaries. Maximum and minimum value must be detected before the comparison stage. Secure protocol choosing max/min number in secure multi-party computation maybe in use.

- **Compression before broadcasting:** In this case, Step 5 of Algorithm 1 should be modified as below:

5: P_j selects the first $\lceil \log_2 m + 1 \rceil$ numbers of c_j as \bar{c}_j . He broadcasts \bar{c}_j and gains the order.

Comparing with compression before calculation, this method is more convenient and efficient. It doesn't increase the complexity but improve its security. Because data slot leaks less information than the whole data. Take case study in Table 1 as example,

$$m = 10, \text{ and } \lceil \log_2 m \rceil = 4.$$

Step 5 publish each one's cross product value as (2340, 2945, 2192, 3059, 2637, 2200, 2519, 2715, 2572, 2796) instead of (23405, 29451, 21925, 30595, 26375, 22007, 25199, 27157, 25723, 27960), and they will get the same sequence as before compression.

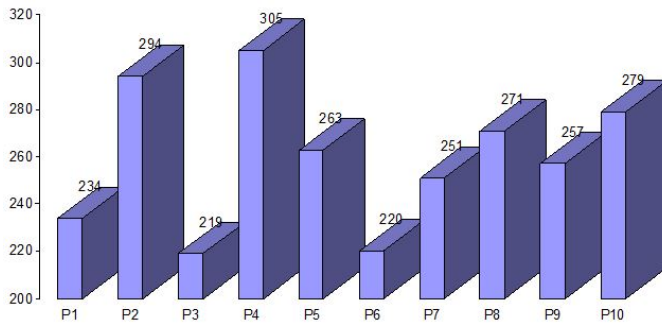


Figure 4: Case of compression before broadcasting

Sometimes, broadcasting only $\lfloor \log_2 m \rfloor$ numbers of c_j will work. For (234, 294, 219, 305, 263, 220, 251, 271, 257, 279) also gets the same sequence in Figure 4.

In some special case, compress method should modified to take the middle number for sorting, like (34, 94, 19, 05, 63, 20, 51, 71, 57, 79) for the case study in Table 1. So, the way of compression before broadcasting can be changed in special environment.

6 Arbitration Procedure

Partner may cheat for sake, he might bring out a false number in the broadcast step. In this case, an arbitration procedure will be in need.

6.1 Third-party Arbitration

If there is a trusted third party who calculates the suspect number honestly, fraud case can be detected easily.

What TTP must verify are two aspects.

- If the calculation result matches the number broadcasted in Step 5.
- If the calculation factors match the candidates' private information.

Although the third-party arbitration is not complex, it is unsafe for partner's sensitive information. A method without TTP is more useful in the social network.

6.2 Privacy-preserving Arbitration

By the help of secure equality-testing and scalar product protocol, the arbitration procedure can be carried out without leaking privacy.

Suppose that P_A suspects that P_B broadcasted a false $\bar{c}_B \neq c_B$ in Step 5, the privacy-preserving arbitration runs as below.

- P_A selects a random number r satisfying $0 < r < 10$.
- P_A holds the $\tilde{E} = (e_1 + r, e_2 + r, \dots, e_n + r)^T$ and P_B holds his privacy D_B . After the secure scalar product

protocol, P_B holds $u = \tilde{E} \cdot D_B + v = \sum_{k=1}^n (e_k + r) \times d_{k,B} + v = \sum_{k=1}^n e_k \times d_{k,B} + r \times \sum_{k=1}^n d_{k,B} + v$ while v is another random number selected by P_A .

- P_A demands P_B to publish his sum $sum_B = \sum_{k=1}^n d_{k,B}$.
- Secure equality-testing protocol tells P_A if $\bar{c}_B + r \times sum_B + v$ is equal to u , it means if P_B broadcasted a false number for cheat.

7 Conclusions and Further Work

Privacy preserving similarity sorting protocol offers a secure solution for problems among m partners in n dimensional systems, which uses secure sum protocol for security and a rational dimensionality reduction for efficiency. After proposing the protocol, we present complexity and security analysis detailed in steps. A case showing computation process is posed in this paper with discussions of some improvements for certain settings and arbitration award for fraud. A further research on secure self-adaption sorting will be considered, while a more reasonable dimensionality reduction method will be brought in for different conditions. In addition, customized version for special use is also considered in the future.

Acknowledgments

The author wishes to thank the fellows in National High Performance Computing Center for helpful comments. And we are very grateful to Professor W. Yang, who is at Suzhou Institute of Advance Study, University of Science and Technology of China, for useful suggestion and some corrections. This work is supported by Funding Project for high tech and industry from Jilin Development and Reform Commission under Grant No. [2014]817.

References

- [1] I. Casas, J. Hurtado, and X. Zhu, "Social network privacy: Issues and measurement," in *Web Information Systems Engineering (WISE'15)*, pp. 488–502, Miami, USA, Nov. 2015.
- [2] B. Dan, S. Laur, and R. Talviste, *Oblivious Sorting of Secret-Shared Data*, Tallinn, Estonia: Institute of Information Security, 2013.
- [3] B. Dan, S. Laur, and R. Talviste, "A practical analysis of oblivious sorting algorithms for secure multi-party computation," in *Secure IT Systems*, pp. 59–74, Tromso, Norway, Oct. 2014.
- [4] W. Du and Z. Zhan, "A practical approach to solve secure multi-party computation problems," in *Proceedings of the 2002 workshop on New security paradigms (NSPW'02)*, pp. 127–135, Virginia Beach, USA, Sept. 2002.

- [5] O. Goldreich, "Secure multiparty computation," *Chapman and Hall CRC*, vol. 2, no. 3, pp. 927–938, 2010.
- [6] K. Hamada, I. Dai, K. Chida, and K. Takahashi, "Oblivious radix sort: An efficient sorting algorithm for practical secure multi-party computation," *IACR Cryptology ePrint Archive*, vol. 2014, pp. 121–150, 2014.
- [7] K. Hamada, R. Kikuchi, I. Dai, K. Chida, and K. Takahashi, "Practically efficient multi-party sorting protocols from comparison sort algorithms," in *Information Security and Cryptology (ICISC'12)*, pp. 202–216, Seoul, Korea, Nov. 2012.
- [8] K. V. Jónsson, G. Kreitz, and M. Uddin, "Secure multi-party sorting and applications," *IACR Cryptology ePrint Archive*, pp. 122, 2011.
- [9] E. Kavakli, C. Kalloniatis, H. Mouratidis, and S. Gritzalis, "Privacy as an integral part of the implementation of cloud solutions," *Computer Journal*, vol. 58, no. 10, pp. 2213–2224, 2014.
- [10] S. Patel, D. Punjani, and D. C. Jinwala, "An efficient approach for privacy preserving distributed clustering in semi-honest model using elliptic curve cryptography," *International Journal of Network Security*, vol. 17, no. 3, pp. 328–339, 2015.
- [11] S. Sarpong, C. Xu, and X. Zhang, "An authenticated privacy-preserving attribute matchmaking protocol for mobile social networks," *International Journal of Network Security*, vol. 17, no. 3, pp. 357–364, 2015.
- [12] S. Sarpong, C. Xu, and X. Zhang, "Ppam: Privacy-preserving attributes matchmaking protocol for mobile social networks secure against malicious users," *International Journal of Network Security*, vol. 18, no. 4, pp. 625–632, 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] C. Yang, T. Chang, and M. Hwang, "A (t,n) multi-secret sharing scheme," *Applied Mathematics and Computation*, vol. 151, no. 2, pp. 483–490, 2004.
- [15] A. C. Yao, "Protocols for secure computations," in *Foundations of Computer Science Annual Symposium*, pp. 160–164, Chicago, USA, Nov. 1982.
- [16] Y. Yao, S. Ning, M. Tian, and W. Yang, "Privacy-preserving judgment of the intersection for convex polygons," *Journal of Computers*, vol. 7, no. 9, pp. 2224–2231, 2012.
- [17] Y. Yao, R. Zheng, and W. Shang, "Privacy preserving outlier detection in social networking services," *Journal of Computational Information Systems*, vol. 9, no. 11, pp. 4299–4307, 2013.
- [18] Y. Zhao, F. Yue, S. Wu, H. Xiong, and Z. Qin, "Analysis and improvement of patient self-controllable multi-level privacy-preserving cooperative authentication scheme," *International Journal of Network Security*, vol. 17, no. 6, pp. 779–786, 2015.

Yifei Yao was born in Jilin Province, China, in 1981. She received the Ph.D. degree from the Department of Computer Science and Technology, University of Science and Technology of China in 2008. She is currently a lecturer of the College of Computer Science and Technology at Changchun Normal University. Her major research interests are information security and distributed computing.

Fanhua Yu received his B.S. degree from the College of Computer Science and Technology, Jilin University in 2004, and Ph.D degree from the School of Transportation, Jilin University in 2008. Now he is a professor and master Tutor of Changchun Normal University. His research interests include artificial intelligence, information security and big data.

An Improved Dual Image-based Reversible Hiding Technique Using LSB Matching

Yu-Lun Wang¹, Jau-Ji Shen¹, Min-Shiang Hwang^{2,3}

(Corresponding author: Min-Shiang Hwang)

Department of Management Information Systems, National Chung Hsing University¹

Department of Computer Science and Information Engineering, Asia University²

No. 500, Lioufeng Raod, Wufeng Shiang, Taichung 41354, Taiwan

Department of Medical Research, China Medical University Hospital, China Medical University³

No.91, Hsueh-Shih Road, Taichung 40402, Taiwan

(Email: mshwang@asia.edu.tw)

(Received Nov. 11, 2016; revised and accepted Jan. 19, 2017)

Abstract

A dual image technique is used as one of the data hiding method. Dual image copies an image to two same images. Through two images to embed or extract secret data, this technique significantly enhances image quality. A dual image technique is good or bad depending on the merits of its algorithm. This paper proposes a method to improve Lu et al. scheme. They use two pixels as a pair and choose two same images to embed, and then choose both two pixels to continue the procedure. We will depend on circumstances of the second pixel pair after previous embedding, and this case can increase the capacity. The experiment results show that our proposed scheme is effective.

Keywords: Data Hiding; LSB Matching; Reversible Hiding Technique

1 Introduction

With the advance of the technology, the speed of the internet becomes faster and faster, and multimedia spreads more easily. Traditional data encryption is through some mathematical operation to encrypt the plaintext into ciphertext [9]. And then the ciphertext is transferred to the receiver via channels. Hence, the ciphertext usually shows a period of distortion. An unauthorized third party can add a period of nonsensical text, so this will cause that the receiver cannot decrypt correctly. Therefore, a data hiding technique was invented [2, 4, 5].

The sender wants to send an image with embedded secret data. In the process of sending an image, the malicious third party is unable to recognize whether an image has embedded secret data or not. After receive the image, the receiver will use an extracted method to extract the secret data. A data hiding technique is also applica-

ble on intellectual property protection of images; through secret data, it can announce the ownership of the images. However, whether image can completely restore or not becomes an important issue. A reversible data hiding technique was invented [8]. Reversible data hiding means that after extracting secret data, we also can get an original image. This is an important technique specially application to the domain which need an original image, such as medical or military images. A data hiding technique has two important criteria: Capacity and quality (PSNR) [1, 3, 6].

This paper will be described as follow. Section 2 will review Lu et al.'s scheme in detail. Section 3 will introduce out proposed method. Section 4 will show our experiment result and analyze Lu et al.'s scheme. Finally, Section 5 will make a conclusion of this paper.

2 Review of Lu et al.'s Scheme

2.1 Embedding Phase

In Lu et al.'s method [10], they will copy an image into two images to process, X and Y . First, they choose $X_i, X_{i+1}, Y_i, Y_{i+1}$ to embed, four bits in each process. First two bits are embedded in first image X , and next two bits are embedded in the second image Y . They use Equation (1) to embed bits at first pixel in both images and to modify the pixel where in some case conditions are met. Then Equation (2) is used to embed bits at second pixel in both images. They also modify the pixels in Table 1.

$$\begin{cases} LSB(P_{i,j}) = \text{embed bit, don't need to change} \\ LSB(P_{i,j}) \neq \text{embed bit, } P_{i,j} - 1 \end{cases} \quad (1)$$

Table 1: Different cases of the embedding phase

Cases	First pixel = embed bit	Second pixel = embed bit	First Pixel change	Second pixel change
1	Yes	Yes	+1	0
2	Yes	No	-1	0
3	No	Yes	0	1
4	No	No	0	0

Table 2: Analysis of the LSB matching method in simultaneously hiding two pairs

Cases	The Pixel value modification statuses				Pixel restoration statuses	
	$X_{i,j}^1$	$X_{i,j+1}^1$	$Y_{i,j}^2$	$Y_{i,j+1}^2$	$P_{i,j}$	$P_{i,j+1}$
1	0	0	-1	0	V	V
2	0	1	0	1		
3	0	1	-1	0	V	
4	-1	0	0	0	V	
5	-1	0	0	1	V	
6	-1	0	-1	0	V	
7	1	0	1	0	V	

$$\left\{ \begin{array}{l} LSB(\lfloor \frac{P_{i,j}}{2} \rfloor + P_{i,j+1}) = \text{embed bit,} \\ \hspace{10em} \text{don't need to change} \\ LSB(\lfloor \frac{P_{i,j}}{2} \rfloor + P_{i,j+1}) \neq \text{embed bit,} \\ \hspace{2em} \text{if first pixel doesn't change, } P_{i,j+1} + 1, \\ \hspace{2em} \text{if first pixel changed, } P_{i,j} + 1 \end{array} \right. \quad (2)$$

In some cases, we need to use a special rule to modify the pixels. These cases are shown in Tables 2, 3, and 4.

Table 3: Modification rule table: Pixel value modification statuses

Rules/Cases	$X_{i,j}^1$	$X_{i,j+1}^1$	$Y_{i,j}^2$	$Y_{i,j+1}^2$
1	0	0	-1	0
2	0	1	0	1
3	0	1	-1	0
4	-1	0	0	0
5	-1	0	0	1
6	-1	0	-1	0
7	0	1	0	0

Assume that we have to embed secret data in four pixels, 100, 97, 91, 110, respectively. And the secret data is 10011111. First, we calculate that LSB (100) of the first pixel of the image X is 0, and the secret data is 1. It is obviously not equal, so we use Equation (1) to get the modified pixel 99. Then we move on the next pixel of the image X and use Equation (2) to calculate whether $LSB(99/2 + 97)$ is equal to the second secret bit 0 or not. The result of this step is equal, so we get the result of the modified pixel is 99 and 97. We keep going to the first pixel of the image Y and calculate $LSB(100)$. Then

Table 4: Modification rule table: The final modified camouflage pixel values

Rules/Cases	$X'_{i,j}$	$X'_{i,j+1}$	$X''_{i,j}$	$X''_{i,j+1}$
1	2	1	-1	0
2	0	1	0	-1
3	2	0	-1	0
4	-1	0	2	1
5	-1	0	2	0
6	-1	2	1	-1
7	-1	-1	1	2

we use Equation (1) to get the modified pixel, 100, and calculate next pixel $Y_{i,j+1}$, by Equation (2). The result of the $Y_{i,j+1}$ is 97. After the procedure we get the modified pixels: 99, 97, 100, and 97, respectively. To check with Table 3, we find that the pixel changing rules are -1, 0, 0, 0, respectively. It conforms to case 4, so we need to further modify $X_{i,j}$, $X_{i,j+1}$, $Y_{i,j}$, $Y_{i,j+1}$, respectively. Finally, we get 99, 97, 102, 98 pixels that represent $X'_{i,j}$, $X'_{i,j+1}$, $Y'_{i,j}$, $Y'_{i,j+1}$, respectively. And then choose next pixel pair to do the embed procedure again, the next pixel pair is 91, 110. We still use Equation (1) to embed secret data in the $X_{i,j}$ and get the modified pixel, 91. Use Equation (2) to embed secret data in the $X_{i,j+1}$ and get the modified pixel, 110. And move on the pixel $Y_{i,j}$, $Y_{i,j+1}$ of the image Y, and, finally, we get the modified pixels are 91, 100, 91, 100, respectively. The pixel changing rule is 0, 0, 0, 0, which doesn't conform to any case in Tables 3 and 4, so we don't need to change the modified pixel. After finishing the embedding phase, send the modified stego-images X and Y to the receiver.

2.2 Extraction Phase

In the extraction phase, we use Equation (3) to extract the first pixel of the pixel pair and Equation (4) to extract the second pixel of the pixel pair. In the recover process, we can use Equation (5) to recover the original cover image.

$$LSB(P_{i,j}) = \text{secret data} \quad (3)$$

$$LSB(\lfloor \frac{P_{i,j}}{2} \rfloor + P_{i,j+1}) = \text{secret data} \quad (4)$$

$$\lfloor \frac{X_{i,j} + Y_{i,j}}{2} \rfloor = \text{original pixel} \quad (5)$$

Assume that we get the modified images X and Y from the sender, we have to extract the secret data and recover the original images. First, we extract first pixel pair of the image X' and Y', 99, 97 and 102, 98, respectively. Then we use Equation (3) to extract first pixel, and LSB (99) is 1. Move on the second pixel of the pixel pair and use Equation (4) to extract the second pixel, and LSB (99/2 + 97) is 0. After extracting the pixel of the image X', we continue to extract the pixel of the image Y', so we still use Equation (3) to extract the first pixel of the pixel pair of the image Y'. LSB (102) is 0 and Equation (4) is used to extract the second pixel of the pixel pair, and LSB (102/2+98) is 1. After extracting process, we can extract secret data 1001 and move on to the next pixel pairs, 91, 110 and 91, 110. We use Equation (3) to extract the first pixel of the pixel pair, and LSB (91) is 1; and then Equation (4) is used to extract second pixel of the pixel pair, and LSB (91/2 + 110) is 1. The extraction result of the pixel pair of the image X' is 11; because the pixel pair of the image X' and the pixel pair of the image Y' are the same, we can extract same secret data 11. The extraction result of the second pixel pair is 1111, so we can correctly extract the secret data just like what the sender embeds in the stego-image. After extraction phase is finished, we can use Equation (5)) to recover the original image. The first pixel in both stego-images X' and Y' are 99 and 102, so we use Equation (5) to recover the first pixel, and (99 + 102)/2 is 100; and after finishing the recover procedure, we can get four pixels as 100, 97, 91, 110.

3 The Proposed Method

In our proposed scheme, the idea of Lu et al.'s method is not always the case, so we will determine whether we use the second pixel of the pixel pair as the first pixel for next embedding phase or not. The condition is as follows: After embedded in two pixels over, we use Equation (6) to decide whether we will use the second pixel or not.

$$|X_{i,j+1} - Y_{i,j+1}| \quad (6)$$

In the embedding phase, we use Equation (1) to embed secret data in the first pixel of the pixel pair and use Equation (2) to embed secret data in the second pixel of the pixel pair. After finishing the embedding process on

the current pixel pair, we use Equation (6) to determine whether we will use the second pixel as the first pixel of the next embedding process or not. If the result of Equation (6) is not equal to zero, we use a new pixel pair to do next embedding process. When the embedding phase is finished, the extraction phase will start after the sender sends the stego-image and the receiver wants to get the correct secret data.

In the beginning, we choose the first pixel pair to extract the secret data, and then we use Equation (3) to extract the first pixel of the pixel pair, and use Equation (4) to extract the second pixel of the pixel pair. After extracting the secret bits correctly, we use Equation (6) to decide whether we will use the second pixel to do next extracting process or not. If the result of Equation (6) is less than 3 or equal to 0, we determine to use the second pixel as the first pixel of the pixel pair of the next extraction process. Otherwise, we use a new pixel pair for next extraction process. And before starting extraction process, we need to recover the second pixel of the pixel pair first; if not, we may not correctly extract the secret bit. Assume that we have four pixels to embed, 100, 97, 91, 110, respectively, and the secret bits is 01011111100. First, we use Equation (1) to embed a secret pixel in the first pixel of image X, and the result of the LSB (100) is 0, which is equal to the secret pixel which we want to embed. Then move on the second pixel of the image X, and use Equation (2) to embed a secret bit; the result of the LSB (100/2 + 97) is 1, which is equal to the secret bit 1. Then after finishing the first process, we get the modified pixels are 100, 97, 100, 97, respectively. We use Equation (6) to determine which pixel will be used in the next embedding process; since 97 - 97 = 0, we use 97 and 91 to do next embedding process; the result of the LSB (97) is 1, and the result of the LSB (97/2 + 91) is 1.

After finishing this embedding process, we use Equation (6) again to decide next embedding process pixels; since 91 - 91 is 0, we use 91, 110 to do next embedding process. The result of the LSB (91) is 1, and the result of the LSB (91/2 + 110) is 1. Then move on the pixel pair of the image Y; LSB (91) is 1 and not equal to the secret bit 0, so the modified pixel is 90. The second pixel LSB (90/2+110) is 1, and is not equal to the secret bit 0, so the modified pixels of image Y is 92, 110, respectively. Finally, we send the modified pixels: 100, 97, 91, 110 and 100, 97, 92, 110 to the receiver.

After the receiver receives the stego-image, we can start the extraction and recovery phases. First we take the first and second pixels of the images X and Y as the pixel pair and use the recover result of the second pixel to extract the secret bit. Starting from image X, the result of the LSB (100) is 0, and the result of the LSB (100/2+97) is 1. And moving on the image Y, the result of the LSB (100) is 0, and the result of the LSB (100/2 + 97) is 1. After finishing the extraction process, we use Equation (6) to determine the pixel pair of the next extraction process. The result of the Equation (6) is 0, so we use 97, 91 and 97, 92 to do next extraction process. The result of LSB

Table 5: Analysis of the improved LSB matching method in simultaneously hiding two pairs

Cases	The Pixel value modification statuses				Pixel restoration statuses	
	$O_{i,j}^1$	$O_{i,j+1}^1$	$O_{i,j}^2$	$O_{i,j+1}^2$	$X_{i,j}$	$X_{i,j+1}$
1	0	0	-1	0	V	V
2	0	1	0	1		V
3	0	1	-1	0	V	V
4	-1	0	0	0	V	V
5	-1	0	0	1	V	V
6	-1	0	-1	0	V	V
7	1	0	1	0	V	V
8	0	0	0	1		V
9	1	0	0	1		V
10	0	1	1	0		V
11	0	1	0	0		V

(97) is 1, and the result of LSB (97/2 + 91) is 1; and the result of the image Y LSB (97) is 1, and the result of LSB (97/2 + 91) is 1. Then we use Equation (6) to calculate 92 - 91 is 1; since it is less than 3 we determine that the pixel pairs of the next extraction are 91, 110 and 92, 110. The result of the LSB (91) is 1, and the result of the LSB (91/2+110) is 1. The result of the pixel pair of the image Y LSB (92) is 0, and the result of LSB (92/2 + 110) is 0. Finally, we extract the secret bits 01011111100 and use Equation (5) to correctly recover the original image. We also propose a special case as shown in Tables 5, 6, and 7.

4 Experiments Result and Analysis

There are two criteria in data hiding area, quality and capacity. We use the grayscale images in the experiment. The source of grayscale images is in the Waterloo Greyscale Set 2 database¹. The images are the TIF format standard images with 512 × 512 pixels. We use a peak signal-to-noise ratio (PSNR) to quantify image quality as follows:

$$MSE = \frac{1}{mn} \sum_{i=0}^{m-1} \sum_{j=0}^{n-1} [P_{i,j} - X_{i,j}]^2$$

$$PSNR = 10 \times \log_{10} \left[\frac{255^2}{MSE} \right]$$

The PSNR value is the higher the better, and the capacity is the bigger the better, where m and n are the images sizes. We use Matlab 8.5.0.197613 (R2015a) and assume that the secret bit which wants to be embedded in the cover image is all of one. The results in Table 8 show that the capacity of the proposed method is better than the original method, and the PSNR is worse than the original method [10].

Table 6: Modification rule table: Pixel value modification statuses

Rule/Case	$O_{i,j}^1$	$O_{i,j+1}^1$	$O_{i,j}^2$	$O_{i,j+1}^2$
1	0	0	-1	0
2	0	1	0	1
3	0	1	-1	0
4	-1	0	0	0
5	-1	0	0	1
6	-1	0	-1	0
7	1	0	1	0
8	0	0	0	1
9	1	0	0	1
10	0	1	1	0
11	0	1	0	0

Table 7: Modification rule table: The final modified camouflage pixel values

Rule/Case	$X'_{i,j}$	$X'_{i,j+1}$	$X''_{i,j}$	$X''_{i,j+1}$
1	2	3	-1	-2
2	0	3	0	-3
3	2	-1	-2	2
4	-1	-2	2	3
5	-1	2	2	-2
6	-1	4	1	-3
7	-1	-3	1	4
8	0	-2	0	3
9	1	-2	0	3
10	0	3	1	-2
11	0	3	0	-2

¹<http://links.uwaterloo.ca/Repository.html>

Table 8: The image and total hidden capacity comparison table

Schemes	Images	Lena	Mandrill	Pepper	Barbara	Boat	Goldhill	Zelda	Washesat
Lu et al. [4]	PSNR(1)	49.13	47.95	49.11	49.14	49	49.17	49.14	49.13
	PSNR(2)	49.12	49.15	49.08	49.11	49.07	49.09	49.09	49.09
	Capacity	524,288	522,996	524,192	524,288	524,208	524,288	524,288	524,276
Proposed Method	PSNR(1)	40.97	40.94	40.99	40.98	40.96	40.98	41.02	41.23
	PSNR(2)	41.30	41.34	41.23	41.20	41.56	41.24	41.03	41.22
	Capacity	617088	618977	608877	610372	632797	613288	599905	614372

5 Conclusion

We proposed an improved method to have a better capacity. The idea of the proposed method is to utilize the current embedded second pixel as the first pixel of next time embedding. However, the quality of the image slightly decreases. How to get both quality and capacity better is the feature work.

Acknowledgments

This research was partially supported by the Ministry Of Science and Technology, Taiwan (ROC), under contract no.: MOST 104-2221-E-468-004 and MOST 105-2410-H-468-009.

References

- [1] L. C. Huang, T. H. Feng, M. S. Hwang, "A new lossless embedding techniques based on HDWT," *IETE Technical Review*, vol. 34, no. 1, pp. 40–47, 2017.
- [2] 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.
- [3] 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, May 2016.
- [4] T. C. Lu, C. Y. Tseng, K. M. Deng, "Reversible data hiding using local edge sensing prediction methods and adaptive thresholds," *Signal Processing*, vol. 104, pp. 152–166, Nov. 2014.
- [5] 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, Mar. 2015.
- [6] 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.
- [7] J. Mielikainen, "LSB matching revisited," *IEEE Signal Processing*, vol. 13, no. 5, pp. 285–287, May 2006.

- [8] J. Tian, "Reversible data embedding using a difference expansion," *IEEE Transactions on Circuits System Video Technology*, vol. 13, no. 8, pp. 890–896, Aug. 2003.
- [9] G. Tychiev, "The encryption algorithm GOST28147-89-PES16-2 and GOST28147-89-RFWKPES16-2," *International Journal of Electronics and Information Engineering*, vol. 6, no. 1, pp. 1–11, 2017.
- [10] S. Zhang, T. Gao, L. Yang, "A reversible data hiding scheme based on histogram modification in integer DWT domain for BTC compressed images," *International Journal of Network Security*, vol. 18, no. 4, pp. 718–727, July 2016.

Biography

Yu-Lun Wang study in the Department of Management Information Systems, Chung Hsing University.

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.

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.