IJNS
International Journal of Network Security

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

Vol. 21, No. 3 (May 2019)

INTERNATIONAL JOURNAL OF NETWORK SECURITY

PUBLISHING OFFICE

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

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

PUBLISHER: Candy C. H. Lin

International Journal of Network Security

Vol. 21, No. 3 (May 1, 2019)

ISCP: An Improved Blockchain Consensus Protocol

Zhong-Cheng Li, Jian-Hua Huang, Da-Qi Gao, Ya-Hui Jiang and Li Fan
(Corresponding author: Jian-Hua Huang)

School of Information Science and Engineering, East China University of Science and Technology
No. 130, Meilong Road, Xuhui District, ShangHai, China
(Email: y30160704@mail.ecust.edu.cn)
(Received Dec. 21, 2017; Revised and Accepted June 15, 2018; First Online Dec. 10, 2018)

Abstract

SCP is a recent effort that focuses on improving the scalability of blockchains by combining PoW and BFT. However, there exist security problems and performance limitation in SCP. In this paper, an improved blockchain consensus protocol ISCP with higher security and better efficiency is presented. We adopt a decentralized multi-partition consensus model to address the security problem in SCP while keeping the computational-scalable feature of the protocol. We also propose a novel intra-committee consensus algorithm which is more efficient than BFT in intra-committee consensus. We analyze and prove that ISCP has higher security and better efficiency than SCP. Experimental results show that the novel intra-committee consensus protocol can significantly reduce consensus delay and greatly increase throughput of the network.

Keywords: Blockchain; Consensus Protocol; Security; Throughput

1 Introduction

Blockchains that can provide trusted, auditable computing in a decentralized network of peers are the underlying technology of cryptocurrency platforms represented by Bitcoin [7]. They also show broad application prospects in fields such as finance, logistics, healthcare [8], and e-commerce [15]. A blockchain is a kind of state machine based on peer-to-peer networks. Ideally, the state of every peer should keep consistent. A Proof-of-Work(PoW) [9] consensus protocol based on CPU power is utilized in the Bitcoin network to achieve consistency among peers by selecting one of participants called miners to issue a proposal that everyone adopts. The miners collect transactions and compete to solve cryptographic puzzles. This process is also known as mining [13]. The PoW consensus can ensure communication efficiency and security of blockchains. However, there are some limitations to the consensus mechanism, such as consuming too much computing power and spending too long time in each epoch. The Bitcoin blockchain grows steadily at a rate of one block every 10 minutes, with size of 1MB per block [12]. The fixed growth speed and block size lead to poor throughput of only 7 transactions per second (Tx/s). Worse still, it will bring about more forks in the blockchain if we simply increase the block size and speed up the generation of blocks, which is likely to result in double-spending [4, 10].

To address the security problem of PoW mechanism under high-speed generating of blocks, Ethereum [2] adopts GHOST (Greedy Heaviest-Observed Sub-Tree) protocol which uses a new policy for selecting the main chain in the block tree to relieve the conflict between security and performance. However, the performance of the performance of GHOST-PoW has not been sufficiently tested. Eyal *et al.* propose the Bitcoin-NG [6] protocol to increase the throughput of blockchains via a primary node appending micro-blocks to the blockchain without proof of work. However, the election of the primary node is based on PoW mechanism which may lead to forks, and the eventual consistency cannot be ensured through this way. Traditional BFT consensus protocols have good performance in throughput, but they require the identities of nodes to be fixed. Furthermore, the communication complexity of BFT will increase dramatically with the increase of participants, so it only works on networks with fewer nodes. Tendermint [1] with low communication overhead is a variant of BFT protocols. It offers better node scalability and security than BFTs.

In recent years, hybrid PoW/BFT consensus protocols become the promising solution for high performance blockchains. SCP [14] is a computationally scalable Byzantine consensus protocol for blockchains. It utilizes a PoW-based identity management mechanism to prevent Sybil attacks [5] and divide nodes into different committees. Moreover, committees generating blocks in parallel through the BFT protocol enables the network throughput to scale approximately linearly with computing power. However, the node scalability of the BFT protocol is poor, and communication complexity increases

dramatically with the increase of nodes within committees, which will lead to long consensus delay. Besides, a final committee is designated to combine the blocks of sub-committees into an ordered blockchain data structure, which may cause the security problem. There exist inherent contradictions between communication complexity and security of the final committee. It is difficult to ensure the security of the protocol while keeping its efficiency. This paper presents an improved SCP protocol (ISCP). We design a decentralized multi-partition consensus model without the final committee to address the security problem in SCP and reduce the communication complexity of the protocol. We further propose a more efficient intra-committee consensus mechanism that simplifies the consensus process and reduces the consensus delay.

The remainder of this paper is organized as follows. In Section 2, we overview some novel blockchain consensus mechanisms that scale PoW and BFT protocols. Section 3 analyses security and efficiency problem of SCP. Section 4 introduces our improved consensus protocol in detail. In Section 5 and Section 6, we analyze the security and efficiency of ISCP thoroughly. Experimental results are presented in Section 7. The contributions of this paper are concluded in Section 8.

2 Related Work

PoW blockchains are not suitable for modern cryptocurrency platforms due to their poor performance. Therefore, many approaches have been proposed to solve the problem. The GHOST protocol used in Ethereum theoretically supports higher throughput than Bitcoin. It adopts a new policy that weights the subtrees rooted in blocks rather than the longest chain rooted in given blocks called the longest chain rule in Bitcoin. The new policy relieves the conflict between performance and security, so it supports higher throughput. However, the performance of GHOST has not been verified yet because the current throughput of Ethereum is only about 0.2 Tx/s on average. Eyal *et al.* proposed Bitcoin-NG that increases network throughput and reduces consensus delay. In Bitcoin-NG, a primary node elected by means of PoW appends multiple micro-blocks that consist of transactions to the blockchain without PoW mining. However, forks will appear during the election of the primary node inevitably and the eventual consistency cannot be guaranteed, which may lead to security problems.

Traditional BFT protocols, which support high throughput, are only applicable in networks with few nodes because they are bandwidth-limited. Classical BFT protocols would run in $O(n^2)$ or $O(n^3)$ communication complexity. With increase of nodes, the consensus delay will eventually become unacceptable. In addition, they cannot tolerate the fluidity of participants. As a result, Byzantine agreement protocols cannot be directly used in blockchain consensus. As a variant of BFT, the Tendermint protocol has higher security, better flexibility than traditional BFTs because participants are forced to lock their coins in a bond deposit during the consensus process and a block is added to the blockchain only if it has been signed by more than 2/3 validators. HoneyBadger [11] is a randomized BFT protocol which supports more nodes than classical BFT protocols and ensures good practical performance. Liu *et al.* [3] argue that the attack model assumed by the BFT systems rarely appears in reality and propose the XFT protocol which reduces the communication complexity and tolerates up to n/2 byzantine nodes simultaneously.

The lightning network proposed by Poon *et al.* [16] increases the transaction throughput through a dedicated fast channel. Through the scalable micro-payment channel network, parties can make high-frequency and bidirectional micro-payment with extremely low delay. However, the security of the lightning network is difficult to guarantee and it essentially belongs to offline blockchain technology. Micro-payment channels [17] increase the throughput of blockchains, but it is also offline blockchain technology and its security is difficult to guarantee.

Hybrid consensus refers to a new kind of consensus mechanism which combines PoW and BFT. SCP is a hybrid consensus protocol using PoW for identity management and BFT for consensus. Generating blocks in parallel enables the throughput of blockchains to scale approximately linearly with the number of participants in SCP.

3 SCP and Its Two-layer Blockchain

SCP utilizes proof-of-work to randomly place nodes into different committees, and these committees propose sub-blocks in parallel, thus improving the throughput of the blockchain network. The problem here is how to combine the outputs of committees into an ordered data structure which will be added to the blockchain. In SCP, a final committee is designated to combine these sub-blocks like the centralized institution. Furthermore, SCP has proved the following lemmas:

Lemma 1. *In every epoch with good randomness, for each committee, at least* $c/2+1$ *committee members will be honest with probability at least* $1 - e^{-27c/160}$. *Moreover, the probability of generating* $c/2 + 1$ *malicious identities by the end of the epoch is also exponentially small.*

Lemma 2. *In every epoch with good randomness, the honest members agree on a unique value with at least* $c/2+1$ *signatures, with probability at least* $1 - e^{-27c/160}$.

Lemma 3. *In every epoch with good randomness, honest members of the final committee will broadcast a combined value (from values from other committees) which has at least* $c/2 + 1$ *signatures, with probability at least* $1 - e^{-27c/160}$.

Where c is the size of each committee, 2^s is the number of committees. Lemma 3 ensures security and correctness of the final committee as long as c is large enough. However, the parameter c has significant influence on efficiency of SCP because the total number of message transmissions is $O(nc + c^3)$ in each epoch where n is the total number of nodes in an epoch. As shown in Figure 1, assuming n is 10,000 (10,510 nodes in Bitcoin and 15,147 nodes in Ethereum until May 2018), the number of message transmissions will reach 390,000 when the size of committee is 35. Moreover, according to Lemma 3, probability that the final committee behaves correctly will decrease dramatically when the size of final committee is below 50, which is shown in Figure 2. Therefore, the correctness of final committee cannot be ensured with overwhelming probability while keeping the efficiency of the protocol.

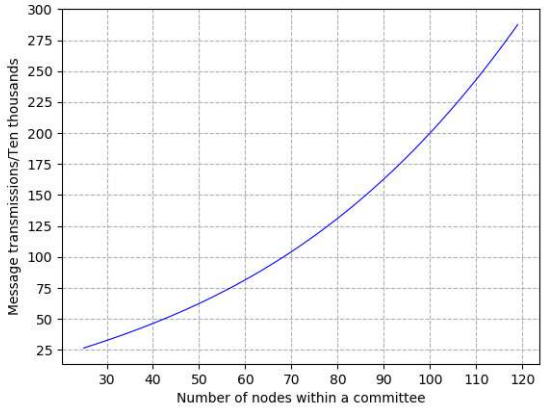


Figure 1: Message transmissions grow polynomial with the size of committee

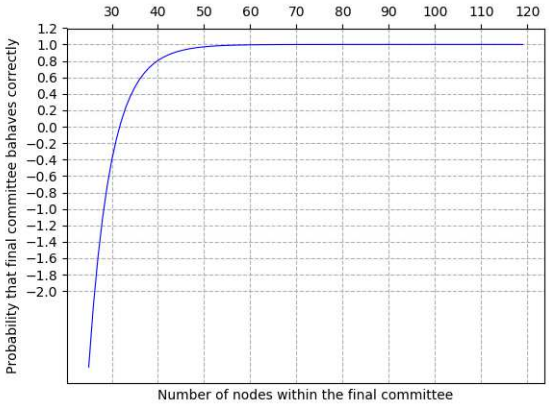


Figure 2: Probability that the final committee behaves correctly decreases dramatically when the size of final committee is below 50

4 ISCP

To address the security and efficiency problems of SCP, we propose ISCP, an improved blockchain protocol. We design a decentralized multi-partition consensus model which consists of only single layer without the final committee. We further propose an inter-committee consensus protocol to ensure all the honest nodes reach an agreement securely and efficiently on the final block which is then added to the blockchain. To further improve the efficiency of ISCP, we also adopt a novel intra-committee consensus algorithm which only requires linear communication complexity.

4.1 Decentralized Multi-partition Consensus Model

As shown in Figure 3, we propose a decentralized multi-partition consensus model. The operation of ISCP is divided into epochs. In each epoch, ISCP splits network participants into several sub-committees to generate sub-blocks in parallel. Similar with SCP, nodes in ISCP are random assigned into different committees according to the PoW computation result. The last r bits of the PoW result is used to specify which committee a node belongs to, *i.e.*, each committee is identified by its r-bit committee id. Unlike SCP, a final committee is not required to integrate sub-blocks in our decentralized multi-partition consensus model. The committees in our protocol are responsible for not only generating sub-blocks but also combining all the correct sub-blocks into a final consensus block.

The consensus process of each epoch is divided into two steps. In the first step, committees run our intra-committee consensus protocol to process separate sets of transactions and generate sub-blocks in parallel. Once a sub-block is verified and signed by at least $c/2 + 1$ members of a committee, the sub-block will be broadcast to all the sub-committees instead of sending to the final committee. In the second step, each committee runs an inter-committee consensus protocol to reach an agreement on the final consensus block that includes all the correct sub-blocks. The final consensus block will be added to the blockchain. At the same time, a random string is revealed to each node to start a new epoch.

4.2 Intra-committee Consensus

4.2.1 Algorithm

The PoW consensus algorithm is designed for Bitcoin networks with a large number of nodes and high mobility, but it is criticized for its heavy computational resource consumption and unstable consensus period. SCP adopts the BFT protocol for consensus in committees. The BFTs protocol, which are bandwith-limited, are not suitable for intra-committee consensus in our system because the number of nodes may exceed 200 (*e.g.* 300) in a single committee. In this paper, we propose a novel intra-

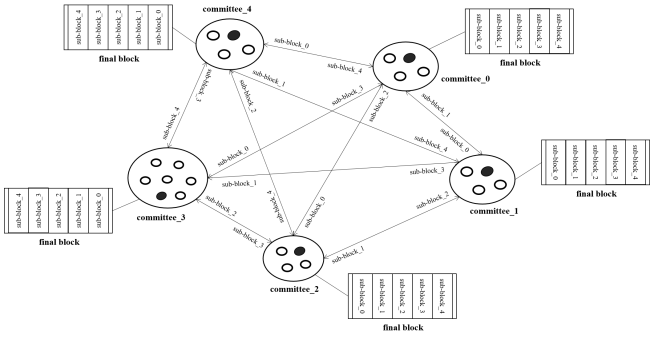


Figure 3: Decentralized multi-partition consensus model

committee consensus algorithm to achieve better consensus performance. We adopt a "retry-on-failure" mechanism to achieve consensus and reduce communication complexity in each committee. The interaction process of the novel intra-committee consensus protocol includes five operation steps. These steps are described as follows:

In Step 1, the leader node in each committee broadcasts the *prepare* message $\langle PreBlockHash, BlockHash, Block_{pre}, random, Signature, CommitteeId\rangle$ within the committee. Where $Block_{pre}$ contains all the correct transactions received by the committee, *random* is a random value chosen by the leader node as the seed for generating *epochRandomness* which will be used in next epoch, and $CommitteeId$ is the identity of the committee;

In Step 2, nodes in the committee verify the correctness of data in $Block_{pre}$, and send a *prevote* message $\langle BlockHash, IP, PK, nonce, Signature\rangle$ to the leader node if the verification succeeds. If a node detects errors in $Block_{pre}$, the node will ask other nodes in the committee to re-elect the leader node.

In Step 3, when the leader node in a partition collects $c/2 + 1$ of the *prevote* messages the for the $Block_{pre}$, it broadcasts a *submit* message $\langle Block, BlockHash, Timestamp, random, PK, Signatures, CommitteeId\rangle$ to the nodes in the committee.

In Step 4, the nodes in the committee verify whether the *Signatures* in the *submit* message contains at least $c/2 + 1$ valid signatures or not. If the verification fails, another leader node will be randomly elected to restart the consensus process. In this paper, a new concept called computation power distance between nodes is introduced to help to randomly elect a new leader node when the current leader node is compromised. We will introduce it later in Section 4.2.2. If the verification succeed, nodes in the committee will broadcast the *submit* message to other committees.

4.2.2 Computation Distance

When the leader node in a committee is compromised, honest nodes will randomly select a new leader node which has the minimal computation distance with the previous leader node to continue the consensus process. In ISCP, we introduce a new concept called computation distance to ensure the randomness of selection. The computation power distance is defined as follows:

$$Dist(node_1, node_2) = Hash_{node_1}\ XOR\ Hash_{node_2}. \quad (1)$$

Where $Dist(node_1, node_2)$ is the computation power distance between $node_1$ and $node_2$, $Hash_{node_1}$ and $Hash_{node_2}$ are the suitable fixed-length hash strings calculated by $node_1$ and $node_2$ in the previous PoW phase. XOR stands for the logical operation whose output is true only when inputs differ. We can easily prove the randomness of the selection because the hash string is randomly generated.

4.2.3 Normal-case Operation

When the leader node in a partition is a non-malicious node, the timing diagram of the protocol is illustrated as Figure 4.

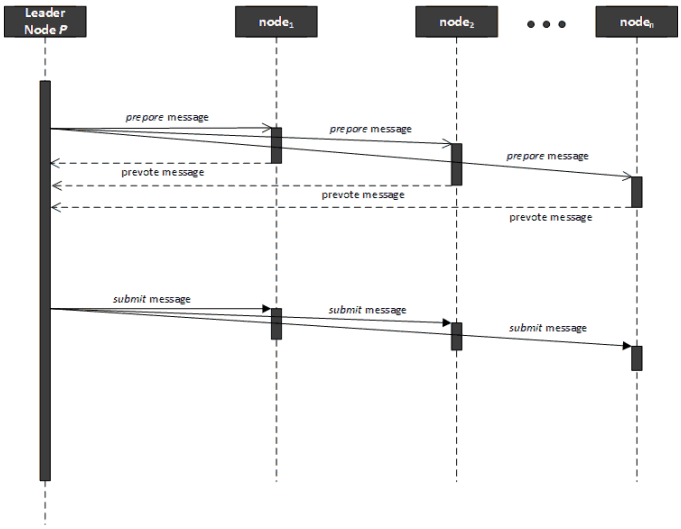


Figure 4: Normal case timing diagram of intra-committee consensus

After joining a committee, leader nodes and ordinary nodes start to collect transactions submitted by users in the blockchain network. In our system, all nodes in a committee can receive transactions and all the received transactions will be broadcasted within the partition. This process ensures that the committee can continue to process transactions even if the leader node is compromised. When the received data reaches a certain number of bytes (such as 1 MB) or the waiting time expires (for example 5 minutes), the committee generates and broadcasts a sub-block following the protocol described above.

4.3 Inter-committee Consensus

After broadcasting sub-blocks, all committees have to achieve consensus on the final block through running the

inter-committee consensus protocol. Our goal is to ensure security with overwhelming probability and achieve $O(n)$ communication complexity which is independent of the size of a committee. An inter-committee consensus protocol is introduced to integrate sub-blocks into a final consensus block. The protocol consists of the following steps:

In Step 1, after receiving a *submit* message with sub-block, an honest node checks whether the sub-block contains at least $c/2+1$ correct signatures or not. If the verification fails, the honest node will discard this message and stop to propagate to other nodes. If the verification succeeds, the honest node will save the sub-block and sends the *submit* message to its neighboring nodes.

In Step 2, once a node has received sub-blocks from all the committee, it begins to take the ordered set union of all transactions in sub-blocks into a final consensus block where sub-blocks are arranged by the order of committee id. If there exist conflicts between sub-blocks, the transaction in the sub-block behind will be deleted from the final consensus block.

In Step 3, a node which has generated the final block becomes a leader node in its committee and the committee run the intra-committee consensus protocol to reach an agreement on the final consensus block.

In Step 4, each committee broadcast its $confirm$ messages $\langle PreBlockHash,\ BlockHash,\ Timestamp,\ CommitteeId,\ epochRandomness,\ nodesList\rangle$ to other committees, where $BlockHash$ is the cryptographic digest of the final consensus block and $epochRandomness$ calculated from seeds in all the sub-blocks is the random value for next epoch of consensus process. $nodeList$ is a list of 20 to 30 members in a committee from where other nodes can download the final consensus block.

In Step 5, once a node has received at least $c/2+1$ valid $confirm$ message with the same $PreBlockHash$ and $BlockHash$, it adds the final consensus block to the blockchain locally and begins the next epoch.

5 Security Analysis

In ISCP, we consider the same threat model and security assumptions as SCP. Malicious nodes may behave arbitrarily and the portion of byzantine adversaries is no more than 1/3. In addition, honest nodes in the network topology are connected and the communication channel is synchronous.

5.1 Intra-committee Consensus Security

As mentioned above, nodes are randomly assigned into different committees, the number of compromised nodes is at most 1/3 at a high probability. We utilize a "retry on failur" method to elect an honest leader node to propose a correct sub-block and reach consensus within a committee. We also adopt a new concept called computation distance to ensure randomness of the election. In each time of election, The probability that the elected leader node behaves arbitrarily is no more than 1/3. In the first x times of elections, the probability P that all the leader nodes are compromised satisfies the following constraint:

$$P = 3^{-x}. \tag{2}$$

As shown in Figure 5, with the increase of election times, the probability that all the previous leader nodes are malicious decreases dramatically and honest nodes in a committee will reach an agreement once an honest leader node turns up. A Malicious leader node may broadcast a sub-block without enough signatures, but honest nodes will refuse to accept it and stop to propagate to other nodes.

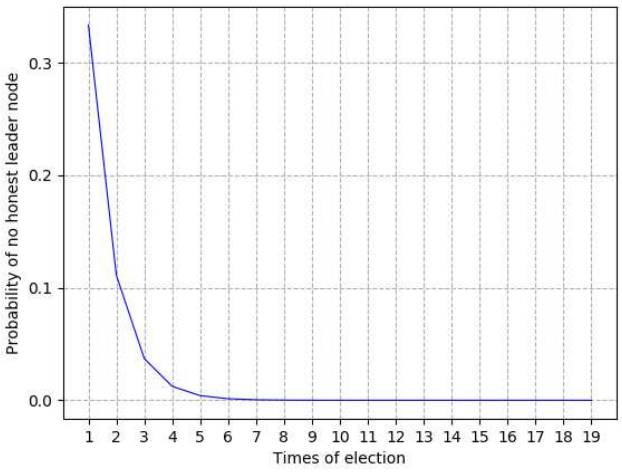


Figure 5: Probability that elected leader nodes are all malicious decreases quickly with times of election

5.2 Inter-committee Consensus Security

During propagation phase of sub-blocks, a malicious adversary may forge sub-blocks to confuse other honest nodes in the network because honest nodes do not know the identities of the nodes in other committees. These counterfeit sub-blocks consist of correct transaction data but are different from the origin ones, *i.e.*, part of transactions are deleted. We will prove that it is extremely hard for malicious adversaries to launch such an attack.

As mentioned above, $Provotes$ in the *submit* message must contain at least $c/2+1$ valid *prevote* messages. The *prevote* message contains IP, PK, $Signature$ and *nonce* of a specific node. Honest nodes can check the validation of an identity by comparing the difficulty and hash string which is calculated from IP, PK and *nonce* in *prevote* message. A malicious adversary wants to forge a sub-block, he must create enough identities to provide enough valid signatures. Moreover, these identities must belong to the same committee which is identified by an

r-bit committee id. The malicious adversary has to search for valid nonce that makes the calculated hash string have (50+r) same bits with the original. If T is the expected time for all the users, collectively, to find one proof-of-work, then the adversary has to take a time T_{byz} to find a satisfied nonce value. The T_{byz} satisfies the following constraint:

$$T_{byz} = 2^s \cdot T. \tag{3}$$

Assuming T10 minutes, 2^s is 32, T_{byz} will be 5.3 hours which are far more than the time an epoch takes. Therefore, It is nearly impossible for the adversary to launch such an attack.

6 Efficiency Analysis

6.1 Intra-committee Consensus Efficiency

In SCP, a committee runs classical consensus protocol such as PBFT to propose sub-block. The number of nodes is c in a committee, the operation of PBFT protocol in normal case is shown in Figure 6. Four phases are needed in each consensus epoch, including *pre-prepare* phase, *prepare* phase, *commit* phase and *reply* phase. During the last *reply* phase, nodes in a committee submit a sub-block to the upper layer (the final committee). The messages required for a consensus process is the sum of messages in four phases, $Msg_{sum} = 2c^2 - c$, and the time complexity is $O(c^2)$.

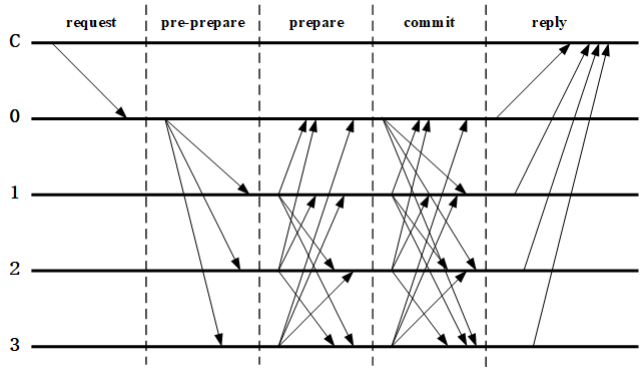


Figure 6: Normal case operation of the PBFT protocol

The operation of our intra-committee consensus protocol in normal case is showed in Figure 7. Four phases are required to complete a consensus, including *prepare* phase in which the leader node broadcasts $Block_{pre}$ to other nodes in the partition for verification, *prevote* phase during which vote messages towards $Block_{pre}$ from other nodes will be sent to the leader node, *submit* phase in which the leader node broadcast a block with enough signatures to the other nodes in the committee, *broadcast* phase in which the committee members broadcast the *submit* message to other committees. The number of messages required for a consensus is $Msg_{sum} = N + c + c + c - 3 = 3c - 3 + N$, and the time complexity is $O(c)$.

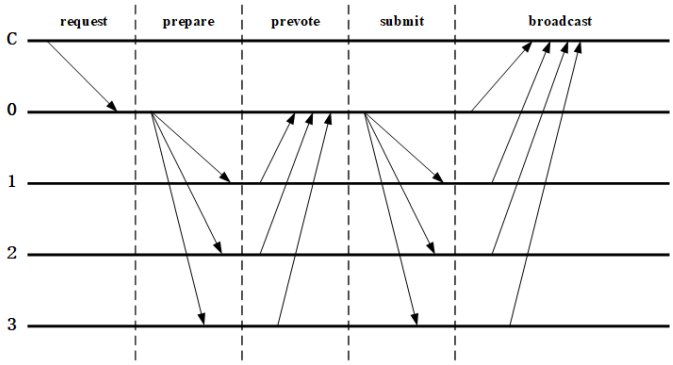


Figure 7: Normal case operation of the intra-committee consensus protocol

PBFT can achieve the state synchronization among honest nodes even if a few nodes are compromised. However, consistency among nodes within a committee is achieved through a voting process in our system. Compared with BFTs, we cancel the mutual communication among the nodes in our intra-committee consensus protocol. Even if the leader node is compromised, other nodes can detect the compromise in time and continue to complete the consensus. From the analysis above, it can be concluded that the intra-committee consensus protocol in ISCP greatly reduces the computation complexity of the protocol compared with the BFTs protocol.

6.2 Inter-committee Consensus Efficiency

In SCP, the number of messages transmitted in the final consensus phase and broadcast phase is $Msg_{sum} = N + c^3$. The final committee has to run the PBFT protocol whenever a sub-block is proposed by a node in committees, which causes very high communication complexity. In contrast, each committee only broadcasts a sub-block to the network in ISCP, which makes the communication complexity independent of the size of committee. The total number of messages transmitted during the inter-committee consensus phase is $(3 + 2^s) \cdot N$ which consists of an intra-committee consensus process and a broadcast of the final block.

7 Experimental Evaluation

Experiments are conducted to test and compare the consensus delay and throughput of ISCP and SCP in the intra-committee.

7.1 Experiment Setup

In the experiments, Docker, an advanced container virtualization technology, is used to simulate network nodes with version of Docker Community Edition 17.09.0-ce-win33 (13620). The codes are based on Python 3. The communication between nodes is based on the UDP protocol. An official Docker image with python version 3.5.4-jessie is the running environment of the codes. The host's memory is 8GB and its operating system is Windows 10 Professional Version 14393.1770. 2048MB memory is allocated to Docker for use.

7.2 Consensus Delay Test

Consensus delay experiments test the time required for SCP and ISCP to complete a consensus in one committee. SCP uses the PBFT protocol to reach a consensus, while ISCP uses the intra-committee consensus protocol reach a consensus. By continuously increasing the number of nodes in the committee, we obtain a delay trend for consensus in a partition, as showed in Figure 8. Each data is the average of 20 test results under the same conditions.

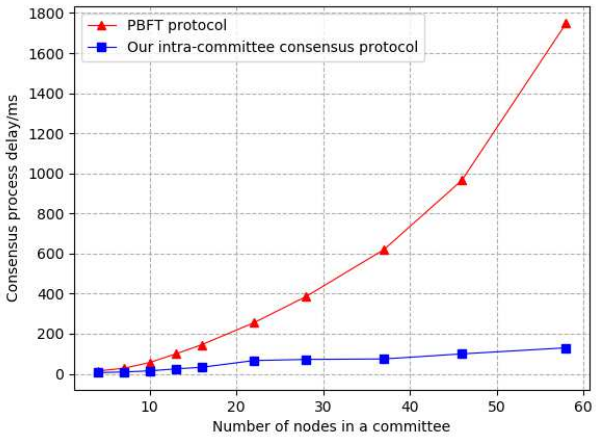


Figure 8: Consensus delay evaluation

In a committee, the consensus delay of SCP approximately grows quadratically with the increase of number of nodes, but the consensus delay of ISCP increases linearly at the same conditions. The reason is that there is many unnecessary communication between nodes in the PBFT protocol used by SCP when the leader node is honest with high probability. Massive message transmission between nodes greatly increases the consensus delay, especially in the internet environment where there may be non-negligible delay during message delivery. The intra-committee consensus protocol in our single-layer blockchain is used for electing a consensus sub-block by voting within a committee. As a result, messages exchanged between nodes are greatly reduced. Experimental results show that our intra-committee consensus protocol can greatly reduce the time it takes to reach consensus in a committee, which enables the committee process network requests more quickly and provide better services.

7.3 Throughput Test

This test compares the processing performance of SCP and ISCP in a committee. SCP uses the PBFT protocol to reach a consensus, while ISCP uses the intra-committee consensus protocol to reach a consensus. A certain number of requests (200 in the test) are send to the committee with sending rate increased constantly. When the sending rate is increased to a certain extent, requests cannot be fully processed in the committee and some messages are lost. We think this is a failure. In the experiment, the failure rate of processing requests of the two protocols is compared at different request sending rate.

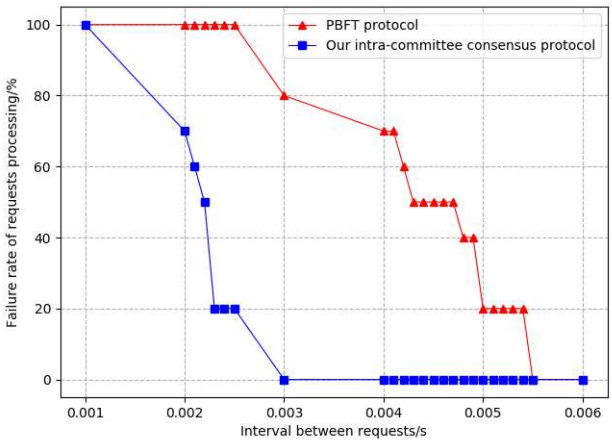


Figure 9: Throughput evaluation

As shown in Figure 9, the failure of the PBFT protocol occurs when requests are processed at a sending rate of 185 requests per second. However, the failure of ISCP occurs at 333 requests per second. At a high level, when new requests arrive at the partition, a queue of requests with limited length is allocated to cache the requests in each member of the partition. If the partition cannot process and remove the requests from the queue in time, the newly arrived requests will be discarded. That is, request processing begins to fail. When a committee in ISCP runs the intra-committee consensus protocol, the delay of consensus process is low and requests are processed quickly. Because the delay of the consensus process of PBFT in SCP are longer than the intra-committee consensus protocol in ISCP, slow request processing rate leads to low throughput. The experimental results show that the committee in ISCP can handle requests with higher sending rate when using our intra-committee consensus protocol. The throughput of the ISCP in committees is higher than that of the SCP committees.

8 Conclusion

BFT used in SCP can result in higher latency and communication complexity in the intra-committee consensus process. In addition, existence of the centralized final committee also leads to the increase of communication complexity and the security of final committee is hard to guarantee, which threatens the system security. This paper introduced ISCP, an improved blockchain consensus protocol to address these problems. We design a decentralized multi-partition consensus model without the final committee and an inter-committee consensus protocol to enable honest nodes to reach an agreement on the final consensus block with high efficiency. We further propose an intra-committee consensus protocol for committee consensus which is more efficient than the BFTs protocol in SCP. The consensus mechanism of ISCP enhanced the performance and security of blockchains. Experimental results showed that the consensus delay of the committees in ISCP is much lower than that of the committees in SCP, especially as the number of nodes increases. The intra-committee consensus protocol of ISCP supports higher processing rates of transactions than PBFT under the same conditions.

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China under Grant 61472139 and a research Grant made to East China University of Science and Technology by Shanghai Education Commission. The authors are also grateful to the anonymous referees for their insightful and valuable comments and suggestions.

References

[1] S. Bano, A. Sonnino, M. Al-Bassam, *et al.* "Consensus in the age of blockchains," *Cryptography and Security*, 2017. (https://arxiv.org/abs/1711.03936)

[2] M. Bartoletti, S. Carta, T. Cimoli, R. Saia, "Dissecting ponzi schemes on ethereum: Identification, analysis, and impact," *Cryptography and Security*, 2017. (https://arxiv.org/abs/1703.03779)

[3] T. Crain, V. Gramoli, M. Larrea, and M. Raynal, *DBFT: Efficient Byzantine Consensus with a Weak Coordinator and its Application to Consortium Blockchains*, Technical Report 1702.03068, 2017. (https://arxiv.org/abs/1702.03068v3)

[4] A. Dmitrienko, D. Noack, M. Yung, "Secure wallet-assisted offline bitcoin payments with double-spender revocation," *ACM*, pp.520-531, 2017.

[5] X. Feng, C. Y. Li, D. X. Chen, *et al.* "A method for defensing against multi-source Sybil attacks in VANET," *Peer-to-Peer Networking and Applications*, vol. 10, no. 2, pp. 305-314, 2017.

[6] A. E. Gencer, S. Basu, I. Eyal, *et al.* "Decentralization in bitcoin and ethereum networks," *Cryptography and Security*, 2018. (https://arxiv.org/abs/1801.03998)

[7] A. Judmayer, N. Stifter, K. Krombholz, *et al.* "Blocks and chains: Introduction to bitcoin, cryptocurrencies, and their consensus mechanisms," *Synthesis Lectures on Information Security Privacy & Trust*, vol. 9, no. 1, pp. 1-123, 2017.

[8] T. T. Kuo, L. Ohnomachado, "ModelChain: Decentralized privacy-preserving healthcare predictive modeling framework on private blockchain networks," *Computers and Society*, 2018. (https://arxiv.org/abs/1802.01746)

[9] J. Li, T. Wolf, "A one-way proof-of-work protocol to protect controllers in software-defined networks," *Symposium on Architectures for Networking & Communications Systems*, pp. 123-124, 2016.

[10] I. C. Lin, T. C. Liao, "A survey of blockchain security issues and challenges," *International Journal of Network Security*, vol. 19, no. 5, pp. 653-659, 2017.

[11] A. Miller, Y. Xia, K. Croman, E. Shi, and D. Song, "The honey badger of BFT protocols," in *Cryptology ePrint Archive*, 2016. (https://eprint.iacr.org/2016/199.pdf)

[12] M. Nofer, P. Gomber, O. Hinz, D. Schiereck, "Blockchain," *Business & Information Systems Engineering*, vol. 59, no. 3, pp. 183-187, 2017.

[13] R. Pass, E. Shi, "Fruitchains: A fair blockchain," *Proceedings of the ACM Symposium on Principles of Distributed Computing*, pp.315-324, 2017.

[14] R. Pass, E. Shi, "Hybrid consensus: Efficient consensus in the permissionless model," *LIPIcs-Leibniz International Proceedings in Informatics*, 2017. (https://eprint.iacr.org/2016/917.pdf)

[15] A. Pazaitis, P. D. Filippi, V. Kostakis, "Blockchain and value systems in the sharing economy: The illustrative case of backfeed," *Social Science Electronic Publishing*, vol. 125, pp. 105-115, 2017.

[16] J. Poon, T. Dryja, "The bitcoin lightning network," *Draft*, 2015. (http://lightning.network/lightning-network.pdf)

[17] E. Rohrer, J. F. Laß, F. Tschorsch, "Towards a concurrent and distributed route selection for payment channel networks," *Networking and Internet Architecture*, 2017. (https://arxiv.org/abs/1708.02419)

Biography

Zhong-Cheng Li had received the B.Eng degree in computer science and technology from East China University of Science and Technology, Shanghai, China. He is currently pursuing the M.Sc. degree in computer science and technology from East China University of Science and Technology, Shanghai, China. His current research interests include blockchain technology and distributed network security.

Jian-Hua Huang had received the B.S. and M.S. degrees from East China University of Science and

Technology, Shanghai,China, and the Ph.D. degree in control theory and control engineering from East China University of Science and Technology, Shanghai,China. He has served as Associate Professor of Computer Science and Engineering at East China University of Science and Technology since 1998. His current research interests include computer networks, wireless sensor networks, information security, data mining, cloud computing, optimization and modeling. Dr. Huang has been a member of various network committees including Specialist Group of Shanghai Education and Research Network and Network Specialized Committee of Shanghai Higher Education Association. He is also the director of Development Center of Shanghai Education Network IPv6 Laboratory.

Da-Qi Gao had received the PhD degree in Industrial Automation from Zhejiang University, China, in 1996. Currently, he is a Full Professor in the Department of Computer Science at East China University of Science and Technology. He has authored or coauthored more than 100 papers. His research interests include Machine Learning, Pattern Recognition, Neural Networks and Artificial Olfactory.

Ya-Hui Jiang had received the B.Eng degree in information security from Jiangsu University, Jiangsu, China. She is currently pursuing the M.Sc. degree in computer science and technology from East China University of Science and Technology, Shanghai, China. Her current research interests include blockchain technology and secure multiparty computation.

Li Fan had received the B.Eng degree in information security from Qingdao University, Shandong, China. She is currently pursuing the M.Sc. degree in computer science and technology from East China University of Science and Technology, Shanghai, China. Her current research interests include blockchain technology and distributed network security.

OCFSII: A New Feature Selection Based on Orthogonal Centroid Both Inter-class and Intra-class for Vulnerability Classification

Jialiang Song[1], Jihong Han[1], Xirui Zhang[1], Lulu Shao[1], and Yang Wang[2]
(Corresponding author: Jialiang Song)

Zhengzhou Information Science and Technology Institute[1]
Zhengzhou 450001, P.R. China
Xi'an Surveying and Mapping Technological Center[2]
(Email: sjl1032011026@163.com, 496443004@qq.com)
(Received Sept. 20, 2017; Revised and Accepted Mar. 27, 2018; First Online Dec. 10, 2018)

Abstract

With the rapid development of information technology, vulnerability has become a major threat to network security management. Vulnerability classification plays a vital role in the whole process of vulnerability management. It is the key point to select proper features to represent categories. Due to the low efficiency and accuracy of some common feature selection algorithms, in this paper, we proposed a new method called OCFSII, which measures the importance of the feature terms both in inter-class and intra-class based on orthogonal centroid. We evaluated the method on the vulnerability database, using two classifiers, namely, KNN and SVM. The experimental results show that the proposed method OCFSII outperforms Information Gain (IG), Document Frequency (DF), Orthogonal Centroid (OC), and is comparable with Improved Gini index (IGI) when KNN used while OCFSII is superior to the four algorithms. In addition, OCFSII is more advanced than OC.

Keywords: Classifier; Feature Selection; Information Security; OCFSII; Vulnerability Classification

1 Introduction

Nowadays, with the development of network technology, people can get information in different channels. To meet the demands of various users, the relevant products, such as different operation systems and application software, are developed, which greatly promotes transmission and sharing of information. However, because of the defects of operation and application software on design or on own disadvantage of programming languages, these products have various disadvantages on design and realization. In terms of information security, the most significant defect is inevitable security vulnerabilities [12]. With the improvement of information society, the coverage rate of Internet devices is improving while the number of security vulnerabilities increases in exponential type. Therefore, it is significant to manage the numerous vulnerabilities [10].

As an important link of vulnerability management, the key point in vulnerability classification is to describe and distinguish different vulnerabilities accurately. Accurate classification of security vulnerability is the basis to continue to analyze and manage vulnerability. It can also greatly help vulnerability researcher know profoundly generation cause and attack influence of the same kind of vulnerability, and it provides key reference information for security administrator to assess severity of vulnerability correctly. Detailed data about vulnerability is indispensable core data for computer security tool and vulnerability classification, while these data are based on text information.

It is the key for vulnerability classification to establish relationship between feature and category and moreover, it is the key point for research of this paper about how to select proper vulnerability features to represent vulnerability category. Whether vulnerability features are proper or not will greatly influence the accuracy of vulnerability classification. In recent years, research popularity for text features selection still increases. Venter *et al.* [13] have put forward a kind of automatic classification scheme based on Self Organization Maps (SOM), which is a kind of data cluster algorithm. The main contribution of this method lies in a type of experimental vulnerability classification model, which does not need to define the vulnerability category manually in advance. It can collect vulnerability samples with similar features into different categories automatically by SOM algorithm. But, this method has low accuracy and efficiency. Mingoti *et al.* [9] has improved vulnerability classification model based on SOM cluster algorithm in [13] using N-Gram replacement word, which has advanced accuracy of cluster. Wang *et al.* [14] has proposed a kind of automatic classification

model for vulnerability based on Bayesian network, which trains Bayesian network by vulnerability information obtained from NVD database and then divides vulnerability into categories defined by CWEs. Chen *et al.* [2] have presented a kind of automatic classification model based on SVM, which trains the SVM classifier with vulnerability information obtained from the CVE list and divides vulnerability automatically into predefined vulnerability features categories. Zhang *et al.* [17] have put forward the research on vulnerability classification method based on fuzzy entropy features selection algorithm. This method can classifies different vulnerabilities combining the advantages of fuzzy entropy theory and SVM classification method and give the evidence for vulnerability features selection to calculate fuzzy entropy. In addition, many scholars have put forward various feature selection algorithms to select more reasonable vulnerability features and improve accuracy and learning ability of vulnerability classification.

However, these methods have some disadvantages. Here are the following points to be improved:

1) Factors for assessing these algorithms are too simple and the situation that distinguishes categories via features is usually considered from one perspective. For example, in [16], document frequency only measures the significance of a feature term in the intra-class while in [1, 15], orthogonal centroid feature selection algorithm and DIA association only calculate the score of a feature in the inter-class. Namely, these algorithms do not take into account importance of features both in the inter-class and intra-class.

2) During the results and analysis of these algorithms, experiments are carried out only by utilizing one same kind of classifier. And the influence in different types of classifiers on accuracy of vulnerability features is not compared. For example, in [4], only naïve Bayes is taken as an experiment tool of vulnerability classification while in [5, 6], only SVM is taken as an experiment tool of vulnerability classification.

3) These algorithms need to obtain vulnerability text resource from vulnerability database, while different vulnerability database has different text factors.

It is necessary to formulate the unified vulnerability text factors to enhance the applicability

To solve the above problems, this paper compares factors in different vulnerability databases, and proposes the standard vulnerability text factors. Moreover, we put forward a new features selection algorithm, called Orthogonal Centroid Features Selection algorithm both in Inter-class and Intra-class (OCFSII). To confirm this method, we use two classifiers including SVM and KNN in vulnerability data, and compare it with four feature selection algorithms including in Information Gain, Improved Gini index, Document Frequency and Orthogonal Centroid. The experiment results show that the proposed method OCFSII outperforms IG, DF OC, and is comparable with IGI when KNN used while OCFSII is superior to IG, DF, OC and IGI when SVM used.

The main contributions of our paper are as follows.

1) This paper gives the standard and unified vulnerability text factors from different vulnerability database.

2) The proposed method measures the significance of a feature term both in inter-class and intra-class.

The remainder of the paper is organized as follows. In Section 2, vulnerability classification principle, feature selection and feature term -classification matrix are briefly reviewed. After that, the proposed method algorithm is presented in Section 3. Experimental setup and Results are included in Section 4 and Section 5 respectively. Finally, the concluding remarks are drawn in Section 6.

2 Related Work

2.1 Vulnerability Classification Principle

Since vulnerabilities from regular vulnerability databases and open vulnerability resource mainly are presented in text form, this paper classifies vulnerabilities via referring to relevant technologies of text classification. Text classification is a process that divides the given text to one or more predefined text categories according to contents [7]. Similarly, Vulnerability classification is a process that classifies the unknown vulnerabilities into predefined vulnerability categories. From the mathematical perspective, vulnerability classification is a special mapping process actually.

This paper gives formalized description for vulnerability classification: Giving a vulnerability text set $D = \{d_1, d_2, \cdots, d_{|D|}\}$ and a vulnerability category set $C = \{c_1, c_2, \cdots, c_{|C|}\}$, where, $|D|$ and $|C|$ represent the number of vulnerability text and vulnerability categories. There is an unknown ideal mapping Φ between vulnerability text set and vulnerability category set:

$$\Phi : D \to C. \tag{1}$$

The purpose of classification learning is to find a mapping model φ that is the most similar to ideal mapping Φ and based on the given assessment function f, the aim of learning is to make Φ and φ fulfilling the following formula

$$Min\left(\sum_{i=1}^{|D|} f(\Phi(d_i) - \varphi(d_i))\right) \tag{2}$$

Generally, the process of vulnerability classification is shown as Figure 1, which includes learning stage and classification stage. Learning stage consists of training process and test process. In order to find the proper parameters for classifying, the feedback mechanism is introduced, which could improve the training results. Classification stage classifies unmarked vulnerabilities by utilizing classifiers ultimately generated in learning stage and vulnerability classification results are output.

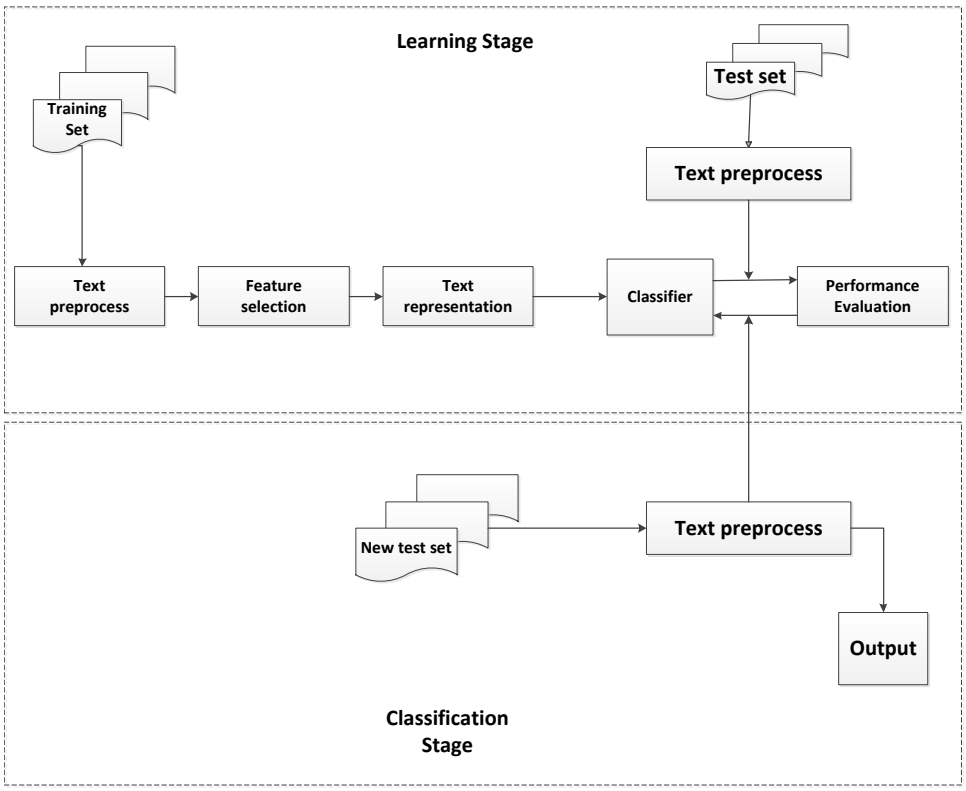


Figure 1: Vulnerability classification process

2.2 Feature Selection

Feature selection is a method which we use proper evaluation criteria to select the optimal features subset from the original feature set. The aim is to select the smallest features subset according to some criteria, so that some tasks, such as classification and regression, achieve better results. Through feature selection, some irrelevant and redundant features are removed, so the simplified data sets often get more accurate models and are easier to understand. In this paper, we give a general framework of feature selection, as shown in Figure 2.

A feature selection algorithm is mainly composed of four parts: generation strategy, evaluation criteria, stop condition and conclusion. The generation strategy refers to generate some feature subsets from the original feature set, while the evaluation criteria means to evaluate the rationality and relevance of feature subsets. Moreover, the stop condition is to determine whether the feature subsets in accordance with initial requirements while conclusion means the validity of feature subsets.

We give the presentation of some popular feature selections including Information Gain, Improved Gini index, Document Frequency and Orthogonal Centroid.

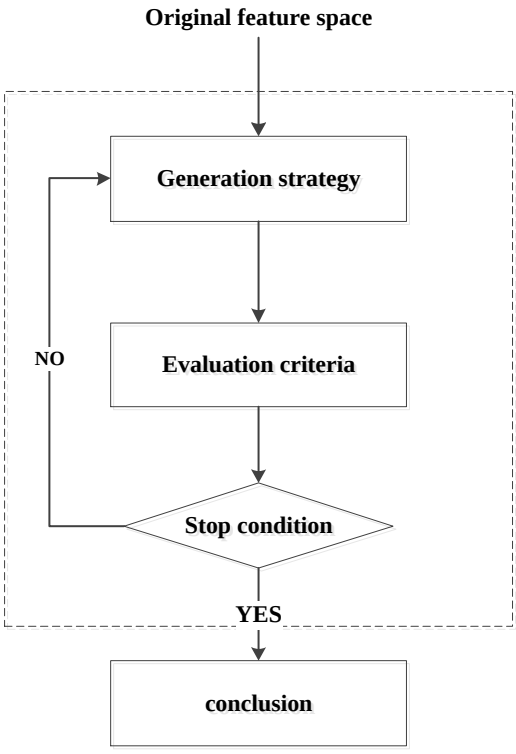


Figure 2: Framework of feature selection

1) Information gain: Information gain is a widely used algorithm in the field of machine learning. The Information Gain of a given feature t_k with respect to the class c_i is the reduction in uncertainty about the value of c_i when the value of t_k is known. The larger Information Gain of a feature is, the more useful the feature is for classification. Information Gain of a feature t_k toward a classification c_i can be defined as

follows:

$$IG(t_k, c_i) = \sum_c \sum_t P(t,c) log \frac{P(t,c)}{P(t)P(c)} \quad (3)$$

where $P(c)$ is the fraction of the documents in category c over the total number of documents. is the fraction of documents in the category c that contain the word t over the total number of documents. $P(t)$ is the fraction of the documents containing the term t over the total number of documents.

2) Improved gini index: Improved Gini index measures the purity of feature t_k toward a classification c_i. The larger the value of purity is, the better the feature is. The formula of the improved Gini index can be calculated as follows:

$$IGI(t_k) = \sum_i P(t_k|c_i)^2 P(c_i|t_k)^2. \quad (4)$$

Where, $P(t_k|c_i)$ is the probability that the feature t_k occurs in category ci. $P(c_i|t_k)$ refers to the conditional probability that the feature t_k belongs to category c_i when the feature t_k occurs.

3) Document frequency: Document frequency is a simple and effective feature selection algorithm that computes the number of documents that contain a feature. The main idea of this algorithm is that if a feature appears in a small number of texts, it is not useful for classification and may even reduce the classification performance. Therefore, the features which possess high document frequency need to be preserved. The formula of Document Frequency can be calculated as follows:

$$DF(t_k, c_i) = P(t_k|c_i). \quad (5)$$

4) Orthogonal centroid: The orthogonal centroid firstly computes the centroid of each category and the whole training set. Then the score of feature is calculated according to the centroid of each class and entire training set. The larger the score of the feature is, the more classification information the feature contains. The formula of orthogonal centroid can be described as follows:

$$OC(t_k) = \sum_{i=0}^{|C|} \frac{n_i}{n} (m_i^k - m^k)^2. \quad (6)$$

Where n_j is the number of documents in the category c_j, is the total number of documents in the training set, m_j^k is the kth element of the centroid vector m_j of category c_j, m^k is the kth element of the centroid vector m of entire training set, $|C|$ refers to the total number of categories in the corpus.

2.3 Feature Term - Classification Matrix

At present, common features selection algorithm is based on Vector Space Model (VSM) and taken into account the property of a features term in a classification, which is called as feature term—classification matrix. In this matrix, row represents feature term in vector space and column represents classification. The property, such as frequency of a feature term in certain classification can be represented by corresponding element value in the matrix. Table 1 shows a feature term—classification matrix, where the value expresses the frequency.

In the table, for example, the frequency of Home in C5 is 111 and other feature terms have low frequency in C5, so Home can represent the C5.

Table 1: Feature term - Classification matrix

Feature term	C1	C2	C3	C4	C5
Home	80	27	11	0	11
Products	5	155	21	0	98
Plan	7	7	0	79	36
Projects	2	0	145	19	1
Design	3	0	6	65	0

3 Unified Description of Vulnerability Factors

Nowadays, different business and institutions possess their own vulnerability database, and the same vulnerability in different database may have different factors. It is not convenient for us to determine which database to choose and which factor to opt. Therefore, it is necessary to formulate the unified vulnerability text factors to enhance the applicability.

3.1 Common Vulnerability Database

CVE is a well-known, widely recognized vulnerability database [3]. Every vulnerability gets a standard name, so it is easy to share data in all kinds of vulnerability database and vulnerability assessment tools. Therefore, we can find the security vulnerabilities of software products more quickly and effectively and give the solution to avoid the threat.

X-FORCE, belonging to ISS, has the most complete the vulnerability items [11]. However, it cannot publish the vulnerability free. ISS offers the online search service.

US-CERT is a middle-class vulnerability database from the Computer Emergency Response Team, and it is built in the Carnegie Mellon University [8]. Also, it can provide online search service.

3.2 Select Unified Factor

The selection of the vulnerability unified factor is the foundation of vulnerability Classification. According to some factors from different database, three institutions selecting the factors for vulnerability classification are shown in Table 2.

Therefore, we select four factors for the unified standards, where the number of the factors selected is three times. It shows that these factors are recognized as the representative attributes in the world. Actually, the factor Date Public is just the time and it has no use for us to vulnerability classification.

Ultimately, we use three factors, CVE name, severity rank and description to express vulnerability. We give an example in Table 3.

4 Algorithm Design

4.1 Algorithm Idea

Orthogonal Centroid Algorithm firstly calculates the centroid of all features in each class and the training set and then calculates the score. We can find that orthogonal centroid algorithm focuses on inter-class, namely calculating the most important feature term compared with other feature terms in one classification. Document frequency is a simple and effective feature selection algorithm. However, Document Frequency method only measures the significance of a feature term in the intra-class. Thus the Document Frequency method concentrates on the column of the feature term-classification matrix while Orthogonal Centroid Algorithm focuses on the row.

Both Document Frequency method and Orthogonal Centroid Algorithm just focus on one respect of the matrix. Therefore, this paper puts forward a kind of new feature selection algorithm, Orthogonal Centroid Features Selection algorithm both in Inter-class and Intra-class (OCFSII), which can make up deficiency of Orthogonal Centroid Algorithm and Document Frequency method and measure comprehensively the importance of a feature term to classification.

4.2 Algorithm Flow

As is shown in Figure 3, we give the flow chart of OCFSII algorithm, which mainly includes two parts, including the construction of feature term-classification matrix and selection of text feature. Text feature selection needs to calculate the centroid of training set. Moreover, we calculate the offset of feature terms both in inter-class and intra-class respectively. Finally, we can obtain the total offset of feature terms and then make a rank for those.

Here, feature term-classification matrix is $V_{T\times C}$, which consists of Tfeatures and Cclasses, matrix element v_{ij} represents the frequency of the ith feature in the jth class, the vector $D = \{d_1, d_2 \cdots d_i\}, 1 \le i \le C$, where d_i represents text number of the ith class. There are some calculation formulas as following:

1) Feature term centroid in training set $M = \{m^1, m^2, \cdots m^i\}$

$$m^i = \sum_{j=1}^{C} v_{ij} / \sum_{j=1}^{C} d_j \tag{7}$$

Where m^irepresents the ith feature term centroid;

2) Feature term centroid in inter-class $M_j = \{m_j^1, m_j^2 \cdots m_j^i\}$

$$m_j^i = \frac{v_{ij}}{d_j} \tag{8}$$

Where, m_j^i represents the centroid of the ith feature term in the jth class;

3) Feature term centroid in intra-class

$$\bar{m} = \frac{\sum_{j=1}^{C} v_{ij}}{C} \tag{9}$$

4.3 Algorithm Description

Algorithm 1 OCFSII algorithm

Input feature term - classification matrix $V_{T\times C}$ and matrix element v_{ij}represents frequency of the ith feature in the jth class; text number vector of class vulnerability$D = \{d_1, d_2 \cdots d_i\}, 1 \le i \le C$; feature number$K$
Output feature subset V_S
$m_i = F_1(v_{ij}, d_i)$ // calculate feature term centroid in training set
$m_j^i = \frac{v_{ij}}{d_j}$ // calculate feature term centroid in inter-class
$\bar{m} = F_2(V_{ij}, C)$ // calculate feature term centroid in intra-class
for $i = 1$ **to** T
    for $j = 1$ **to** C
        $a_{ij} = v_{ij} - \bar{m}$ // calculate offset in intra-class
        $b_{ij} = m_j^i - m^i$ // calculate offset in inter-class
    end for
    $OCFSII_{ij} = a_{ij} * b_{ij}$
end for
$V_S = OCFSII_{TOPK}$

Moreover, the function F1 and F2 are defined as follows:

OCFSII algorithm measures the importance of features both in inter-class and intra-class. And it is so simple to implement. The time complexity is O(T*C) - namely the product of the number of rows and columns in Feature Term-classification Matrix.

Table 2: Institution selecting the factors for vulnerability classification

ID	Factor	CVE	X-FORCE	US-CERT	Times
1	CVE name	✓	✓	✓	3
2	Data public	✓	✓	✓	3
3	Date-up	×	×	✓	1
4	Severity rank	✓	✓	✓	3
5	Credit	×	×	✓	1
6	Solution	×	×	✓	1
7	Description	✓	✓	✓	3

Table 3: CVE-2015-1611 information

CVE name	description	Severity rank
CVE-2015-1611	OpenFlow plugin for Daylight before Helium SR3 allows remote attackers to spoof the SDN topology and affect the flow of data, related to fake LLDP injection.	Middle (CVSS score: 5.0)

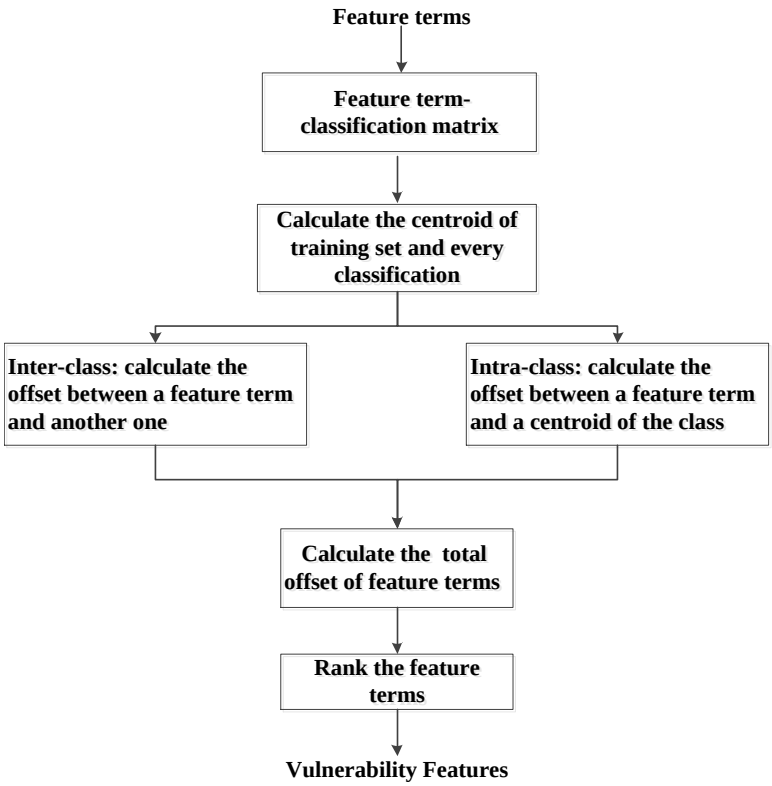


Figure 3: Flow chart of OCFSII algorithm

5 Experiment Setup

5.1 Experimental Classifier

In this section, K-Nearest Neighbor (KNN) and Support Vector Machine (SVM) are described briefly. Both of them are supervised learning method.

1) KNN classifier: KNN classification is a kind of learning algorithm based on sample, which is considered as an inert method. This algorithm shows wonderful performance in many applications. The key point of this method is to find a proper similarity measure to determine the degree of similarity between sample and training set. Therefore, we can get the nearest training set from the unmarked samples.

2) SVM classifier: SVM is a kind of machine learning algorithm, which is widely used in machine learning.

Algorithm 2 $F_1(v_{ij}, d_i)$

$v_{ij} = 0; d_j = 0$
for $i = 1$ **to** T
  for $j = 1$ **to** C
    $v_{ij} = v_{ij} + 1$
    $d_j = d_j + 1$
    $m_i = \frac{v_{ij}}{d_{ij}}$
  end for
end for
return (v_{ij}, d_j)

Algorithm 3 $F_2(v_{ij})$

$v_{ij} = 0$
for $i = 1$ **to** T
  for $j = 1$ **to** C
    $v_{ij} = v_{ij} + 1$
    $\bar{m} = \frac{v_{ij}}{C}$
  end for
end for
return (v_{ij})

Moreover, SVM is a high efficient classifier in classification. In our study, we choose liner kernel SVM.

5.2 Experimental Data

The purpose of this experiment is to select the feature of vulnerability, so as to verify the accuracy and efficiency of the vulnerability classification. In addition, we do not give a profound study on the selection of the categories. The vulnerabilities are divided into the most common six categories, authentication, buffer errors, cross-site scripting, code injection, information leak and input validation respectively. The sample set of vulnerabilities is shown in Table 4.

We select 3500 vulnerabilities from Security Content Automation Protocol (SCAP), from which 3000 vulnerabilities belong to training sample and 500 vulnerabilities belong to test training. As is seen from the Table 4, 3000 samples will train the classifier alter the feature selection and, the 500 samples are utilized to test the accuracy of OSFCII.

5.3 Experimental Steps

In this section, we give the concrete the steps of vulnerability classification experiment.

1) Obtain the original vulnerability features via preprocessing the vulnerabilities text from the database;
2) Construct the feature term- class matrix;
3) Get the feature terms of each category by utilizing the proposed feature selection OCFSII;
4) Use VSM to quantify the vulnerability feature terms;
5) Utilize one-to-many method to structure the vulnerability classifier;
6) Calculate the F1 and accuracy and give the experiment results.

5.4 Performance Measures

In our experiment, we utilize the F1 and Accuracy to measure the performance of the vulnerability classification.

1) Precision and micro-precision: Precision is the ratio of the number of vulnerability texts which are correctly classified as the positive class to the total number of those which are classified as the positive class. The formula of the precision for class c_i is defined as:

$$P_i = \frac{TP_i}{TP_i + FP_i} \tag{10}$$

Where TP_i is the number of vulnerability texts which are correctly classified as class $c_i$and FP_i means the number of vulnerability texts which are misclassified as class.

Similarly, in order to evaluate the performance average across the classes and micro-precision is used in this paper. The formula of the micro-precision can be calculated:

$$P_{micro} = \frac{TP}{TP + FP} = \frac{\sum_{i=1}^{|C|} TP_i}{\sum_{i=1}^{|C|} (TP_i + FP_i)} \tag{11}$$

Where $|C|$ is the number of the classes.

2) Recall and micro-recall: Recall is the ratio of the number of vulnerability texts which are correctly classified as the positive class to the total number of those which are actually belong to the positive class. The formula of the precision for class c_i is defined as:

$$R_i = \frac{TP_i}{TP_i + FN_i} \tag{12}$$

Where FN_i means the number of vulnerability texts belonging to class c_i are misclassified to other classes.

Similarly, in order to evaluate the performance average across the classes and micro-precision is used in this paper. The formula of the micro-precision can be calculated:

$$R_{micro} = \frac{TP}{TP + FN} = \frac{\sum_{i=1}^{|C|} TP_i}{\sum_{i=1}^{|C|} (TP_i + FN_i)} \tag{13}$$

3) F1 and accuracy: When we obtain the micro-precision and micro-recall, the formula of the F1 and Accuracy can be calculated:

$$F_1 = \frac{2P_{micro}R_{micro}}{P_{micro} + R_{micro}} \tag{14}$$

Table 4: Experimental vulnerability sample

Category	Number		
	Training sample	Testing sample	The total
authentication	221	30	251
buffer errors	303	80	383
cross-site scripting	945	200	1145
code injection	830	110	940
information leak	250	30	280
input validation	451	50	501

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (15)$$

Where TN means the number of vulnerability texts which are correctly classified to other classes excluding the positive class.

6 Results

6.1 Experimental Results when KNN Classifier

Table 5 shows the F1 measure results when KNN classifier is used. In this chart, we can see that OCFSII actually has the best performance when the number of features is 300, 500, 1300 and 1500. Moreover, the IGI has the similar performance compared with OCFSII but the latter is superior to the former. All of the algorithms have the positive correlation when the number of the features between 300 and 1300 and from 1300 on, the F1 measure begins to decrease. Therefore, the number of the features is 1300 for the database and the experiments can get the excellent results.

Similarity, Figure 4 shows the accuracy measure results when KNN classifier is used. In this graph, OCFSII and IGI have the better results compared with other methods. And all the curves rise from 300 to 1300 and decline later. It proves that when the number of features is 1300, and we can get the good results. OCFSII has the greatly improved when we consider the importance of features both in inter-class and intra-class compared with OC.

6.2 Experimental Results when SVM Classifier

Table 6 shows the F1 measure results when SVM classifier is used. In this chart, it can be seen that F1 measure results when utilized OCFSII outperforms any other methods. Although IGI has the similar performance compared with OCFSII, the latter precedes the former a little. Similarly, all of the algorithms have the positive correlation when the number of the features between 300 and 1300 and from 1300 on, the F1 measure begins to decrease. So, we can select 1300 features for the database approximately to obtain the good results. Compared with KNN

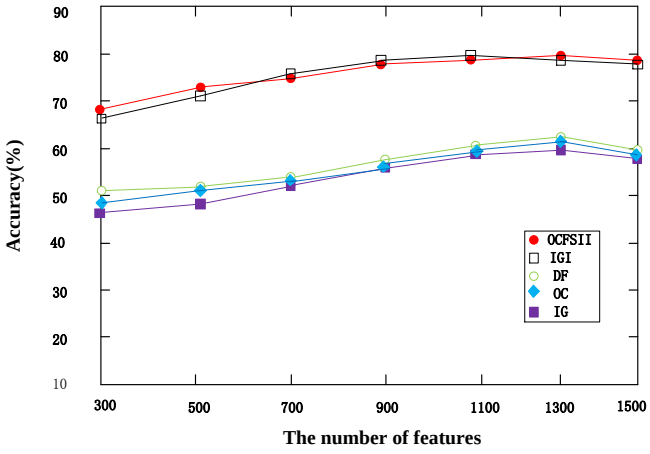


Figure 4: Accuracy measure curve using KNN classifier (%)

classifier used in the experiment, SVM classifier performs better when we use the same methods.

Similarity, Figure 5 shows the accuracy measure results when SVM classifier is used. In this graph, OCFSII has the better results compared with other methods. All of the curves ascend gradually with the increasing of the number of features, and they reach the highest point when the number is 1300. It tells us that we can get the excellent performance when we select 1300 features approximately. Obviously, OCFSII has the greatly improved compared with OC because both inter-class and intra-class are taken into consideration. Compared with KNN classifier used in the experiment, SVM classifier outperforms when we use the same methods.

7 Conclusion

In order to protect the information and network from the numerous numbers of the vulnerabilities, it is significant to manage the vulnerabilities. Classification, as a key link of vulnerability management, plays a major role in this whole process. Due to some feature selection method just consider the importance of the feature term from one aspect, we proposed a new feature selection algorithm called

Table 5: F1 measure results using KNN classifier (%)

The number of feature	300	500	700	900	1100	1300	1500
OCFSII	69.22	72.54	74.65	76.70	77.76	78.55	77.21
IG	47.88	49.43	52.12	55.23	57.12	58.21	57.33
DF	50.32	52.88	54.76	57.45	59.23	60.52	59.21
IGI	68.54	71.21	74.87	76.92	77. 99	78.32	77.10
OC	49.83	51.43	53.43	56.32	58.22	59.43	58.45

Table 6: F1 measure results using SVM classifier (%)

The number of feature	300	500	700	900	1100	1300	1500
OCFSII	71.54	73.85	75.81	78.60	79.76	80.21	79.21
IG	50.39	51.66	53.94	56.43	57.99	58.32	57.10
DF	52.47	54.18	56.85	58.21	60.23	61.76	60.53
IGI	70.35	72.31	75.32	78.10	78.54	79.21	78.29
OC	51.43	53.57	55.22	57.47	59.22	60.25	59.64

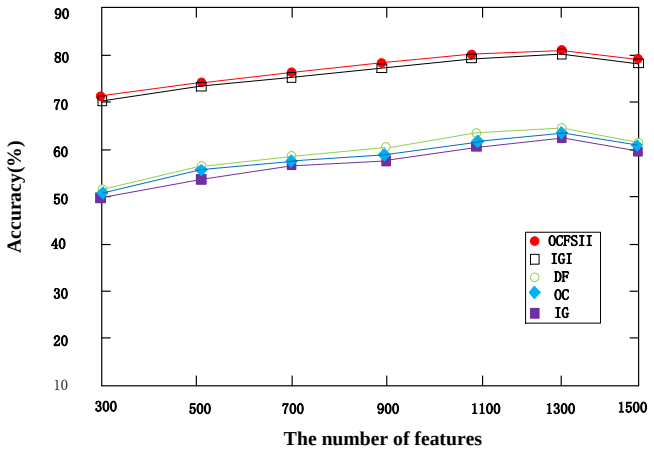


Figure 5: Accuracy measure curve using SVM classifier (%)

OCFSII, considering the importance of the feature term both in inter-class and intra-class. To confirm the validity of this method, we use two classifiers including SVM and KNN in our experiment, and compare it with four feature selection algorithms including Information Gain, IGI, Document Frequency and Orthogonal Centroid. The experiment results show that the proposed method OCFSII outperforms IG, DF OC, and is comparable with IGI when KNN used while OCFSII is superior to IG, DF, OC and IGI when SVM used. As part of our future research, we plan to design the better method to improve the accuracy and efficiency to enhance the understanding of vulnerability essence. [8]

References

[1] B. Bigi, "Using kullback-leibler distance for text categorization," *Lecture Notes in Computer Science*, vol. 2633, pp. 305–319, 2016.

[2] Z. Chen, Y. Zhang, and Z. Chen, "A categorization framework for common computer vulnerabilities and exposures," *The Computer Journal*, vol 53, no. 5, pp. 551-580, 2010.

[3] S. Christey and R. A. Martin, "Vulnerability type distributions in CVE," *The MITRE Corporation*, 2007. (`http://cwe.mitre.org/documents/vuln-trends/index.html`)

[4] A. K. Gupta and N. Sardana, "Naive bayes approach for predicting missing links in ego networks," in *IEEE International Symposium on Nanoelectronic and Information Systems*, pp. 161–165, 2017.

[5] K. M. A. Hasan, M. S. Sabuj, and Z. Afrin, "Opinion mining using naive bayes," in *IEEE International Wie Conference on Electrical and Computer Engineering*, pp. 511–514, 2016.

[6] H. J. Kim, J. Kim, and J. Kim, "Semantic text classification with tensor space model-based naive bayes," in *IEEE International Conference on Systems, Man, and Cybernetics*, pp. 004206–004210, 2017.

[7] H. Lodhi, C. Saunders, J. Shawe-Taylor, N. Cristianini, and C. Watkins, "Text classification using string kernels," *Journal of Machine Learning Research*, vol. 2, no. 3, pp. 419–444, 2002.

[8] P. Minarik and T. Dymacek, "Netflow data visualization based on graphs," in *Proceedings of Visualization for Computer Security, International Workshop*, pp. 144–151, 2008.

[9] S. A. Mingoti and J. O. Lima, "Comparing som neural network with fuzzy -means, -means and traditional hierarchical clustering algorithms," *European*

Journal of Operational Research, vol. 174, no. 3, pp. 1742–1759, 2006.

[10] 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.

[11] F. H. Schmitz, "Reduction of blade-vortex interaction (BVI) noise through x-force control," *Journal of the American Helicopter Society*, vol. 43, no. 1, pp. 14–24(11), 1995.

[12] J. Song, J. Han, D. Zhang, L. Yuan, and L. Shao, "Evaluation of security vulnerability severity based on cmahp," in *IEEE International Conference on Computer and Communications*, pp. 1056–1060, 2017.

[13] H. S. Venter, J. H. P. Eloff, and Y. L. Li, "Standardising vulnerability categories," *Computers & Security*, vol. 27, no. 3-4, pp. 71–83, 2008.

[14] J. A. Wang and M. Guo, "Vulnerability categorization using bayesian networks," in *Proceedings of the Sixth Annual Workshop on Cyber Security and Information Intelligence Research (CSIIRW'10)*, pp. 1–4, 2010.

[15] J. Yan, N. Liu, B. Zhang, S. Yan, Z. Chen, Q. Cheng, W. Fan, and W. Y. Ma, "Ocfs: Optimal orthogonal centroid feature selection for text categorization," in *International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 122–129, 2005.

[16] Y. Yang and J. O. Pedersen, "A comparative study on feature selection in text categorization," in *Fourteenth International Conference on Machine Learning*, pp. 412–420, 1997.

[17] P. Zhang and X. Y. Xie, "Research on vulnerability classification based on svm with fuzzy entropy feature selection algorithm (in Chinese)," *Application Research of Computers*, vol. 32, no. 4, pp. 1145-1148, 2015.

Biography

Jialiang Song received the B.S. degree in electronic science and technology from Zhengzhou information science and technology institute, Henan, China, in 2015. He is pursuing the M.S. degree in information security at Zhengzhou information science and technology institute. His research interests include vulnerability management and information security.

Jihong Han received the B.S., M.S. and Ph.D degrees in information security from Zhengzhou information science and technology institute, Henan, China, in 1983, 1990, and 2008, respectively. She is now a professor at the Department of Information Security, Zhengzhou information science and technology institute. Her research interests include information hiding, watermarking and software reliability. She has published 60 research articles and 3 books in these areas.

Xirui Zhang received the B.S. degree in management from Zhengzhou information science and technology institute, Henan, China, in 2016. He is pursuing the M.S. degree in information security at Zhengzhou information science and technology institute. His research interests include information security and data mining.

Lulu Shao received the B.S., M.S in information security from Chongqing communications college, Chongqing, China, in 2004 and 2011. She is pursuing the Ph.D degree in information security at Zhengzhou information science and technology institute. Her research interests include wireless communication security.

Yang Wang received the B.S. degree in computer science and technology from Henan University, China, in 2014. He is pursuing the M.S. degree in computer science and technology at Zhengzhou information science and technology institute. His research interests include software reverse analysis and network security.

Additively LWE Based Homomorphic Encryption for Compact Devices with Enhanced Security

Ratnakumari Challa[1] and Gunta Vijaya Kumari[2]
(Corresponding author: Ratnakumari Challa)
Department of Computer Science, Engineering, Rajiv Gandhi University of Knowledge Technologies[1]
IIIT-AP, RKValley, Kadapa Dist, Andhra Pradesh, India
Department of Computer Science, Engineering, Jawaharlal Nehru Technological University, Hyderabad[2]
Kukatpally, Hyderabad, Telangana, India
(Email: ratnamala3784@gmail.com)
(Received Nov. 03, 2017; Revised and Accepted July 7, 2018; First Online Dec. 10, 2018)

Abstract

LWE based homomorphic encryption scheme has been proven as secured technique and consequently widely implemented since the last decade. However, the scheme is not considered to be practical because of its larger size of public keys and ciphertexts. In this paper, LWE based additively homomorphic encryption technique is further modified to enhance the performance in minimizing the space required for public keys and ciphertext without compromising the security of the scheme. This makes the scheme suitable for implementation in low space devices. The practical implementation of the scheme is explored and its performance analysis has been presented here. The scheme is also compared with the standard LWE.

Keywords: Compact Devices; Learning with Errors; Prime Factoring; Pseudorandom Generator

1 Introduction

A Fully Homomorphic Encryption (FHE) is treated as the*holy grail* of cryptography, because, it allows arbitrary processing of data in encrypted mode [8, 13]. An encryption scheme is called homomorphic if, given two ciphertexts say $c_1 = E_k(m_1)$ and $c_2 = E_k(m_2)$ where m_1, m_2 are plaintexts and k is the key, one can compute $c = c_1 o c_2 = E_k(m_1 o m_2)$ for some operation o such that $D_k(c) = m_1 o m_2$ [11]. This idea of homomorphic encryption was initially proposed by Rivert, et al [19] in 1978. If o corresponds to a single operation of either addition or multiplication only, the scheme is said to be partially homomorphic. Several partially homomorphic encryption schemes were proposed and successfully used in applications such as electronic voting, private information retrieval, multiparty computation, oblivious polynomial evaluation and so on [11, 15]. However, to perform arbitrary computations over the encrypted data so that the scheme is suitable for any application in general, it must support both addition and multiplication operations over the ciphertexts without any limits in case of which it is called as fully homomorphic encryption (FHE) [8]. Solving a three-decade old long lasting cryptographic problem of designing an FHE scheme was a dream of cryptographers, which was first theoretically solved in a pioneering work by Craig Gentry in 2009 using an innovative construction method [8] Gentry's FHE is based on the algebraic lattice theory and consists a general blueprint that can be used for the construction of FHE schemes. However, the scheme was practically infeasible due to high computational complexities underlying the blueprint, specifically, the bootstrapping or ciphertext refreshing process. In a quest for devising a practical FHE scheme, several variants of the Gentry's scheme were explored [20–22]. Many new schemes based on different security assumptions and hard algebraic and number theoretic problems such as Approximate Greatest Common Divisors (AGCD) [22], Chinese Remainder Theorem (CRT) [5], identity based [13] and attribute based schemes [10] were proposed. Though all these works have shown progressive improvements one over the other, none of them could be a candidate for practical deployment. Therefore, devising an FHE scheme with practical time complexities is still an open problem.

The Learning with errors (LWE) based cryptographic scheme was first proposed and implemented by Regev [18] in 2009. LWE problem has been considered well suited for new research on public key cryptography. LWE based cryptosystem is proven as simple and fast for implementation. Moreover, the security of the LWE problems is proven to be hard [17] since it is related to the well-known "learning parity with noise" problem.

Several variants of LWE based Homomorphic encryption schemes such as FHE using standard LWE [2, 3], RLWE (Ring LWE) [2, 14], and other variants [1, 4, 9]

have been proposed. The theoretical implementation of the schemes has been considered to preserve privacy in cloud computing while practical implementation of LWE based homomorphic encryption is considered to be complex due to its larger public keys and ciphertexts. The space constraint also limits the implementation of these schemes on compact devices. However, there were some techniques [6, 7, 12, 16] proposed elsewhere to reduce the size of public keys and ciphertexts which are suitable for the implementation in low speed and low storage devices. The aim of this work is to propose the possible implementation of the LWE based cryptosystem suited for the devices with low storage capacity.

Contributions. The major contribution of this paper is to reduce the size of the public keys and ciphertexts. The scheme is theoretically presented with time complexities for the Encryption, Decryption, Key Generation and Additions functions in the previous work [16]. In this work, the extended version of the encryption is presented to further minimize the public key space without compromising the security.

2 Preliminaries

2.1 Notations

In this section, basic concepts related to LWE notations are presented for quick appreciation of the proposed work. An integer is denoted as small case letter in single quotations (*e.g.*, $'n'$). The bold upper case letters (*e.g.*, $\boldsymbol{V}$)) are used to denote vectors. The symbols $'+'$ and $'.'$ are used for the addition and multiplication operations respectively. The symbol $< \boldsymbol{V}_1, \boldsymbol{V}_2 >$ denotes the integer which resulted as the sum of individual products of elements of the vectors $\boldsymbol{V}_1$ and $\boldsymbol{V}_2$.

2.2 LWE Based Homomorphic Encryption

Standard LWE based homomorphic encryption scheme [3] is constructed based on two major parameters: $'n'$ (dimension) and $'q'$ (modulus). First Secret vector $\boldsymbol{S}$ of $'n'$ integers is chosen and then set of public keys $(\boldsymbol{A}_i, b)$, denoted as $PK = \{PK_1, PK_2...\}$, is computed using key generation function, where $\boldsymbol{A}_i$ is an arbitrary vector of $'n'$ integers, and $'b'$ is an integer computed as $< \boldsymbol{A}_i, \mathbf{S} > +2.e_i$; where $'e_i'$ is small randomly chosen error. Now, for every j^{th} public key generation, the random arbitrary vector $\boldsymbol{A}_j$ is chosen, and j^{th} public key, PK_j, computed from the secret key $\mathbf{S}$ as follows: $PK_j = (\boldsymbol{A}_j, b_j) = (\boldsymbol{A}_j, < \boldsymbol{A}_j, \boldsymbol{S} > +2.e_j)$.

Given the plaintext message bit m_i, encryption algorithm computes the ciphertext $\boldsymbol{C}_i = (\boldsymbol{A}_i, v_i)$ using any random j^{th} public key $PK_j = (\boldsymbol{A}_j, b_j)$ where $v_i = b_j + m_i \pmod q = (< \boldsymbol{A}_j, \boldsymbol{S} > +2.e_j + m_i \pmod q)$ and $\boldsymbol{A}_i = \boldsymbol{A}_j$.

Decryption takes ciphertext $\boldsymbol{C}_i = (\boldsymbol{A}_i, v_i)$ and computes the plaintext message bit $'m_i'$ using the secret key $\boldsymbol{S}$. The decryption process is given as follows:

$$m_i = (v_i - < \boldsymbol{A}_i, \boldsymbol{S} >) mod 2,$$

Decryption eliminates two masks and leaves the message bit as output.

3 Scheme with Shorter Public Keys and Ciphertexts

In this section, we formally present the LWE based additively homomorphic encryption scheme with shorter public keys and ciphertexts. Seed based technique is proposed to minimize the each public key as well as ciphertext from $(n+1).log_2(q)$ bits down to $2.log_2(q)$ bits [7, 16]. A pseudo random number generator with initialized seed value $(seed_j)$ is used to generate the vector $\boldsymbol{A}_j$ of 'n' elements which in turn generates the j^{th} public key $PK_j = (\boldsymbol{A}_j, b_j)$.

Instead of publishing the public key $PK_j = (\boldsymbol{A}_j, b_j)$ of size $n+1$, the public key vector is shortened to two integers and published as $(seed_j, b_j)$. LWE based encryption with shortened public keys with the support of homomorphic encryption [16] is formally presented as follows:

KeyGen function. It takes initial parameters, modulus $'q'$ and dimension $'n'$ as inputs, then it chooses the secret key vector $\boldsymbol{S}$ of $'n'$ integers and generates set of public keys $PK(= \{PK_1, PK_2, ...\})$. Any i^{th} element of PK, using parameters $'n', 'q'$ and the secret key vector $\boldsymbol{S}$, is computed as follows:

1) Choose a prime value $'p_i'$ as a seed value and pass it to the pseudo random number generator for generating $'n'$ integers of vector $\boldsymbol{A}_i$.
2) Compute the public key $PK_i = (\boldsymbol{A}_i, b_i)$ as $(\boldsymbol{A}_i, < \boldsymbol{A}_i, \boldsymbol{S} > +2e_i)$ where $'e_i'$ is a small random error.
3) Publish the public key $PK_i = (p_i, b_i)$ instead of $(\boldsymbol{A}_i, b_i)$.

Encryption function. For encrypting any i^{th} plain text message bit m_i (0 or 1)

1) Choose any k^{th} public key $PK_k : (p_k, b_k)$
2) Compute ciphertext $\boldsymbol{C}_i$ using public key PK_k as follows:

$$\boldsymbol{C}_i = (p_i, v_i) = (p_k, b_k + m_j).$$

Homomorphic addition function. Given any two ciphertexts $\boldsymbol{C}_x = (p_x, v_x)$ and $\boldsymbol{C}_y = (p_y, v_y)$, compute the new sum cipher $\boldsymbol{C}_z$ as follows:

$$\boldsymbol{C}_z = (p_z, v_z),$$

where $p_z = p_x.p_y$ and $v_z = v_x + v_y$.

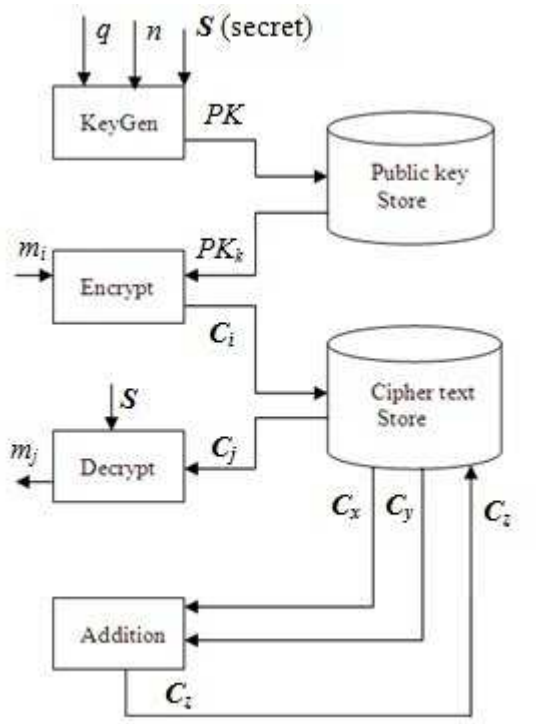


Figure 1: Proposed LWE based additively homomorphic encryption process

Decryption function. For any given ciphertext $\boldsymbol{C}_j = (p_j, v_j)$ decryption function computes the plaintext message bit m_j using the secret key $\boldsymbol{S}$. The process is given as follows:

1) Compute set of prime factors p_1, p_2, p_3 and so on from p_j using prime factoring technique.
2) Generate vector $\boldsymbol{A}_i$ of size $'n'$ using pseudo random number generator function from the seed value $'p'_i$ for all prime factors computed from $'p'_j$.
3) Compute sum vector $\boldsymbol{A}$ from all vectors $\boldsymbol{A}_1$, $\boldsymbol{A}_2$, $\boldsymbol{A}_3$,... as follows:
$$A = \sum_i A_i$$
4) Compute message m_j from ciphertext using the following relation,
$$m_j = (v_j - < \boldsymbol{A}, \boldsymbol{S} >) \bmod 2$$
where $\boldsymbol{S}$ is the secret key vector.

The whole process of key generation, encryption, and decryption and addition operation is illustrated as shown in the Figure 1.

4 An Extended Encryption

The key generation function is used to generate a set of $'x'$ public keys and these keys are stored in the public store.

In the encryption process, a public key, $PK_k = (p_k, b_k)$ is chosen randomly from the public key store to encrypt the plain text message bit m_i and to compute the ciphertext $\boldsymbol{C}_i = (p_i, v_i)$ as described under Encryption Function in Section 3. The ciphertext thus computed is stored in publicly accessed ciphertext store. Also, the second component $'v'_i$ of the ciphertext is computed by adding the message bit (0 or 1) to the second component $'b'_k$ of the public key, PK_k. If the message is 0, then $v_i = b_k$ and if the message is 1, then $v_i = b_k + 1$. This makes very little difference to the component $'v'_i$ of the ciphertext. Since the public key store and the ciphertext store are publicly accessible, an adversary can choose any ciphertext from the ciphertext store and search for its corresponding public key (used for its encryption) in the public key set. If the size of public key set is small, then it becomes easy for an adversary to search for its corresponding public key, hence, to compute the message bit. Therefore, big size of the public key set ensures that the scheme is secure. However, the huge size may affect the suitability of scheme in implementing over compact devices.

Modification in the encryption function, proposed earlier [16], helps to reduce the number of keys down to small number and to make the scheme suitable for compact devices without compromising the security. The proposed extended version of the scheme utilizes the modified encryption function.

Modified encryption function. For encrypting any i^{th} message bit m_i (0 or 1)

1) Choose any k^{th} public key $PK_k : (p_k, b_k)$
2) Compute ciphertext $\boldsymbol{C}_i$
$$\boldsymbol{C}_i = (p_i, v_i) = (p_k, b_k + m_i + 2.e_i)$$
where $'e'_i$ is randomly chosen error.

Now, the newly added error term 'e'_i in the encryption seems to make the scheme significantly more secure; further, the public key set can be minimized to fit for the low storage devices.

Security of the scheme.

- The security of the scheme is totally dependent on modulus $'q'$, dimension $'n'$ and the secret vector $\boldsymbol{S}$. Hence, its hardness is equivalent to that of the LWE problem.
- It is important to consider the privacy of the operations on the ciphertext as it is important for homomorphic encryption scheme. The size of the ciphertext is reduced to two and it maintains the privacy even after performing the many addition operations on it.

5 Results and Discussion

In the proposed scheme, the total size required for storing $'x'$ public keys or ciphertexts is minimized to $[2x.\log_2(q)]$,

whereas the size required in standard LWE schemes is $[x.(n+1).\log 2(q)]$. The comparison of the storage (in bits) required for publishing 1024 public keys in standard LWE scheme and the proposed scheme is given in Table 1. For the parameter $n = 10$ and for an integer $'q'$ with 10 bits the maximum storage required for 1024 public keys in the proposed scheme is 24477 bits, whereas for the standard LWE scheme, the storage required is 112624 bits. A key implication of this minimization is that the size of the public keys or ciphertexts becomes independent of the security parameter $'n'$.

In the practical implementation of the proposed LWE based additively homomorphic encryption scheme, it is observed that the execution time for the encryption operation is in the same range as that of the previously proposed work [16]. The comparison of the time complexities of standard LWE scheme [3] and the proposed version of the LWE scheme for different values of $'n'$ and $'q'$ are given in Figure 2. It is observed that the execution time for the homomorphic encryption functions of the proposed LWE are comparable to the standard LWE. The execution time for the Key Generation, Encryption and Decryption operations are in the same range too as that of the standard LWE. However, the execution time for the addition operation is significantly low in the proposed LWE which becomes more prominent at higher $'n'$ and $'q'$ values. At $n = 100000$ and $'q'$ to be a 50 bit number, the execution time for the addition operation in standard LWE is 2.1×10^8 whereas for the proposed LWE, it is 4×10^3.

6 Conclusions

An efficient LWE based additively Homomorphic Encryption has been proposed and explored with practical implementation. The prime number used as the seed value for the pseudo random generator helps shorten the public keys and ciphertexts. The proposed enhanced encryption function provides the enhanced security and further minimization of public key space large extent. In the practical implementation of the scheme, the execution time complexities for every function are reasonably small even for higher values of the security parameters. This makes the scheme suitable to implement over compact devices.

Acknowledgments

The authors gratefully acknowledge the anonymous reviewers for their valuable comments.

References

[1] S. Agrawal, D. M. Freeman, and V. Vaikuntanathan, "Functional encryption for inner product predicates from learning with errors," in *Proceedings of The 17th International Conference on the Theory and Application of Cryptology and Information Security (ASIACRYPT'11)*, pp. 21–40, Dec. 2011.

[2] Z. Brakerski, C. Gentry, and V. Vaikuntanathan, "(leveled) fully homomorphic encryption without bootstrapping," in *Proceedings of The 3rd Innovations in Theoretical Computer Science Conference (ITCS'12)*, pp. 309–325, Jan. 2012.

[3] Z. Brakerski and V. Vaikuntanathan, "Efficient fully homomorphic encryption from (standard) LWE," in *Proceedings of The IEEE 52nd Annual Symposium on Foundations of Computer Science (FOCS'11)*, 2011. DOI: 10.1109/FOCS.2011.12

[4] Z. Brakerski and V. Vaikuntanathan, "Fully homomorphic encryption from ring-lwe and security for key dependent messages," in *Proceedings of The 31st Annual Conference on Advances in Cryptology (CRYPTO'11)*, pp. 505–524, Aug. 2011.

[5] J. H. Cheon, J. S. Coron, J. Kim, M. S. Lee, T. Lepoint, and M. Tibouchi, "A batch fully homomorphic encryption over the integers," in *Proceedings of The Advances in Cryptology (EUROCRYPT'17)*, pp. 315–335, 2013.

[6] D. H. Duong, M. P. Kumar, and M. Yasuda, "Efficient secure matrix multiplication over lwe-based homomorphic encryption," *Tatra Mountain Mathematical publication*, vol. 67, pp. 69–83, 2016.

[7] S. D. Galbraith, *Space-efficient Variants of Cryptosystems based on Learning with Errors*, 2013. (`https://www.math.auckland.ac.nz/~sgal018/compact-LWE.pdf`)

[8] C. Gentry, "A fully homomorphic encryption scheme," *ACM Digital Library*, 2009. ISBN: 978-1-109-44450-6

[9] C. Gentry, S. Halevi, and V. Vaikuntanathan, "A simple bgn-type cryptosystem from LWE," in *Proceedings of The EUROCRYPT*, pp. 506–522, 2010.

[10] C. Gentry, A. Sahai, and B. Waters, "Homomorphic encryption from learning with errors: Conceptually-simpler, asymptotically-faster, attribute-based," in *Proceedings of TheAdvances in Cryptology (CRYPTO'13)*, pp. 75–92, 2013.

[11] I. Jabbar and S. N. Alsaad, "Design and implementation of secure remote e-voting system using homomorphic encryption," *International Journal of Network Security*, vol. 19, no. 5, pp. 694–703, 2017.

[12] R. Lindner and C. Peiker, "Better key sizes (and attacks) for lwe-based encryption," in *Proceedings of The Cryptographers Track at the RSA (CT RSA'11)*, pp. 319–339, 2011.

[13] L. Liu and J. YeA, "Homomorphic universal reencryptor for identity-based encryption," *International Journal of Network Security*, vol. 19, no. 1, pp. 11–19, 2017.

[14] V. Lyubashevsky, C. Peikertand, and O. Regev, "On ideal lattices and learning with errors over rings," in *Proceedings of The EUROCRYPT*, pp. 1–23, 2010.

[15] D. Rappe, "Homomorphic cryptosystems and their applications," *ResearchGate*, 2004. DOI: 10.17877/DE290R-15728

Table 1: Storage required for 1024 public keys at different security levels

	Parametres		Storage space in bits	
Security Level	n	q **is an integer of**	**Standard LWE scheme**	**Proposed scheme**
Toy	10	10 bits	112624	24477
Small	100	20 bits	2068323	40957
Medium	1000	30 bits	30748377	61436
Large	10000	40 bits	409609747	81914
Very Large	100000	50 bits	5121138606	102422

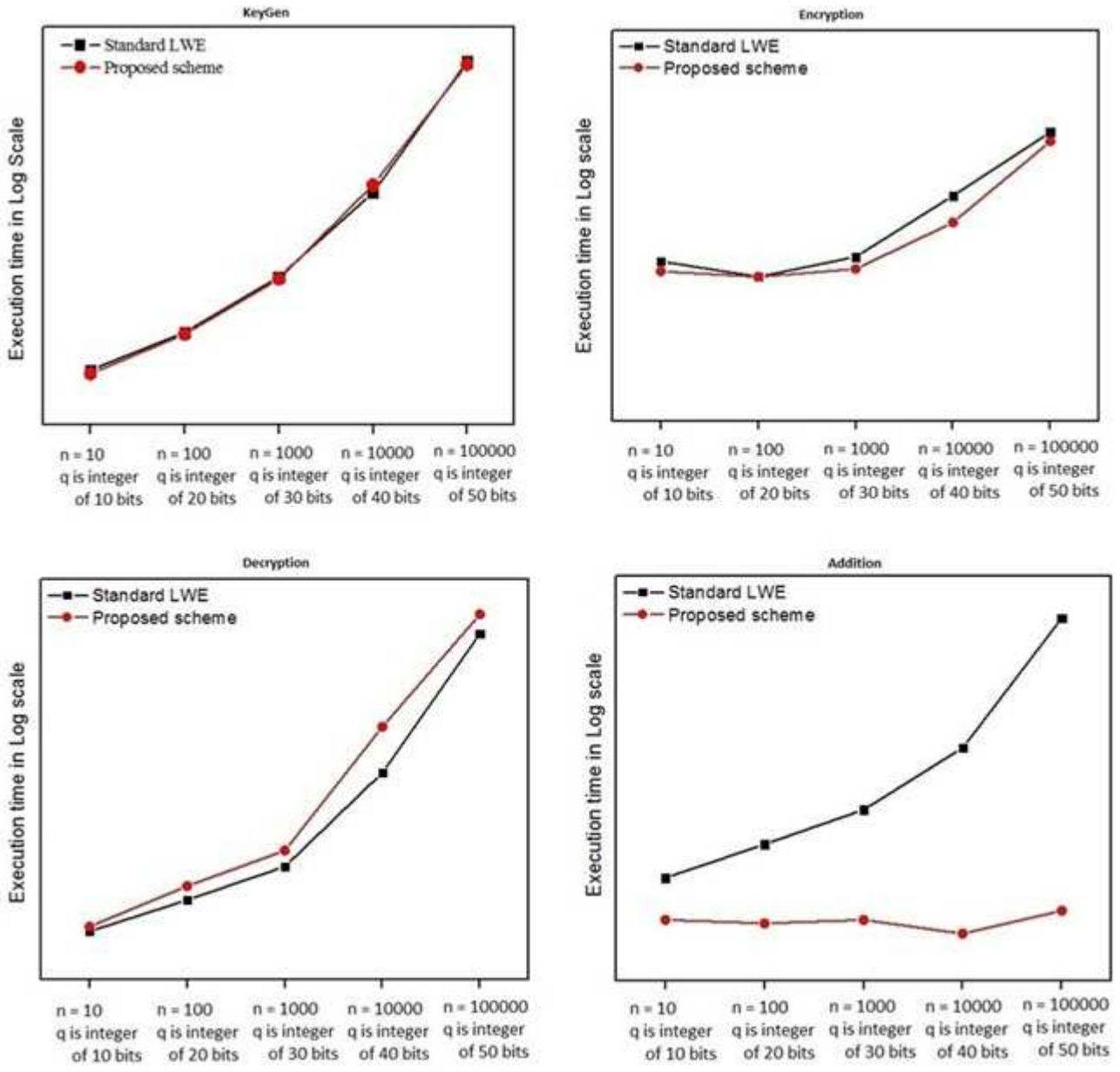


Figure 2: Practical performance of key generation, encryption, decryption and addition functions of the schemes (standard and proposed) at different security levels

[16] C. Ratnakumari and G. VijayaKumari, "An efficient LWE-based additively homomorphic encryption with shorter public keys," in *Proceedings of The Progress in Intelligent Computing Techniques: Theory, Practice, and Applications (ICACNI'16)*, pp. 171–177, 2018.

[17] O. Regev, "On lattices, learning with errors, random linear codes, and cryptography," *Journal of the ACM*, vol. 56, no. 6, 2009.

[18] O. Regev, "The learning with errors problem (invited survey)," in *Proceedings of The IEEE 25th Annual Conference on Computational Complexity (CCC'10)*, pp. 191–204, June 2010.

[19] R. Rivest, L. Adleman, and L.M. Dertouzos, "On data banks and privacy homomorphisms," *Academic Press, Massachusetts Institute of Technology, Cambridge, Massachusetts*, vol. 56, no. 6, pp. 169–180, 1978.

[20] N.P. Smart and F. Vercauteren, "A fully homomorphic encryption with relatively small key and ciphertext sizes," in *Proceedings of The Public Key Cryptography (PKC'10)*, pp. 420–443, 2010.

[21] D. Stehle and R. Steinfeld, "Faster fully homomorphic encryption," in *Proceedings of The Advances in Cryptology (ASIACRYPT'10)*, pp. 377–394, 2010.

[22] M. VanDijk, C. Gentry, S. Halevi, and V. Vaikuntanathan, "A fully homomorphic encryption over the integers," in *Proceedings of The Advances in Cryptology (EUROCRYPT'10)*, pp. 24–43, 2010.

Biography

RatnaKumari Challa biography. She has received her M.Tech degree in Computer Science from University of Hyderabad, India in 2009. She is an Assistant Professor in Department of Computer Science and Engineering, RGUKT, IIIT-AP, Andhra Pradesh, India. She is currently pursuing Ph.D in JNTUH, Hyderabad, India. Her research interests include Security, Cloud Computing and Image processing.

VijayaKumari Gunta biography.She has received her Ph.D degree from University of Hyderabad, India in 2002 . She is a professor in Department of Computer Science and Engineering, JNTUH, Hyderabad. India. Her research interests include Algorithms, Security and Cloud Computing.

An Improved Ternary Montgomery Ladder Algorithm on Elliptic Curves over GF(3^m)

Shuang-Gen Liu, Rong-Rong Wang, Yun-Qi Li, and Can-Liang Zhai
(Corresponding author: Shuang-Gen Liu)

School of Communication and Information Engineering, Xi'an University of Posts and Telecommunications
Xi'an 710121, China
(Email: liusgxupt@163.com)

(Received Jan. 15, 2018; Revised and Accepted Apr. 21, 2018; First Online Jan. 11, 2019)

Abstract

In this paper, we propose a new scalar multiplication algorithm on elliptic curves over GF(3^m). It combines original Montgomery ladder algorithm and the ternary representation of the scalar, which makes full use of cubing. In addition, in order to improve performance, we have presented new composite operation formulas which are $2P_1+P_2$ and $3P_1$, and applied them to the improved scalar multiplication algorithm. Based on the original Montgomery ladder algorithm, it can resist Simple Power Attack (SPA). In the experimental analysis, we set the ratio of inversion and multiplication to a dynamic value. Results show that with respect to previous algorithm, the average efficiency of proposed scalar multiplication algorithm is increased by 7.8 % to 11.3 % in affine coordinate, and 4.0 % to 17.9 % in projective coordinate.

Keywords: Characteristic Three; Composite Formulas; Elliptic Curve Cryptography; Montgomery Ladder Algorithm; Scalar Multiplication

1 Introduction

Elliptic Curve Cryptography (ECC) was introduced independently by Koblitz [9] and Victor Miller [16] in around 1985. Its key size is smaller than RSA with the equivalent security, for example, elliptic curve with the key of 160 bits is competitive with RSA with the key of 1024 bits. This can be especially an advantage for applications where resources are limited, such as smart cards, embedded devices and mobile phones. Its safety is based on the difficulty of elliptic curve discrete logarithm problem (ECDLP). ECC is used for encryption, decryption, digital signature and verification [5, 7, 8, 18, 22, 28]. The factors that affect the execution rate of ECC algorithm are generally as follows: The choice of coordinates, scalar multiplication, the selection of elliptic curves. One of the decisive factors is the calculated rate of scalar multiplication. Scalar multiplication, defined as $[k]$P=P+P+...+P, where k is an integer and P is an elliptic curve point, is a major and time-consuming operation in ECC. Scalar multiplication operations can be divided into two layers: the top layer and the bottom layer. Among them, the top layer operation is basic point operation on elliptic curve, such as point addition and point doubling, and the bottom layer operation is underlying field operation, such as inverse, multiplication, squaring, and so on.

In recent years, GF(2^m)-ECC and GF(p)-ECC have been well studied. There are a lot of methods to improve the elliptic curve scalar multiplication, such as double-and-add [11], non-adjacent form (NAF) [20] and so on, while GF(3^m)-ECC have been less studied due to their efficiency factors. So the research of fast and security scalar multiplication on the elliptic curve over GF(3^m) has become one of the hot research topics. GF(3^m)-ECC, as a special type of GF(p^m)-ECC, has some properties of fast computation similar to GF(2^m)-ECC, but it has own special properties and is suitable as a carrier of secure password algorithm [12, 19, 24, 26].

In 1987, Montgomery [17] proposed a fast algorithm for calculating the elliptic curve scalar multiplication kP that resists Simple Power Attack(SPA) attacks. SPA is a type of side channel attack proposed by literature [10]. The basic idea is: Integer k is expanded into binary form, which is computed cyclically from left to right. And there are one point doubling and one point addition operations in each cycle. Because of the same computational pattern and cost in every loop iteration, this algorithm prevents SPA. However, the original Montgomery ladder algorithm has the demerit of slow performance. In this paper, a scalar k is represented as ternary form instead of binary form. The length of the ternary expression is shorter than binary expression. Based on Montgomery's idea, a new algorithm was proposed by Lopez and Dahab [15] in 1999, and it was used for calculating the elliptic curve scalar multiplication over GF(2^m). By using a new set of point addition and point doubling calculation formulas, every iteration requires only the x-coordinate of the point to be calculated, and the y-coordinate is restored

at the end of the algorithm. Smart and Westwood [27] first pointed out that ordinary elliptic curves over finite fields of characteristic three is an alternative for implementing elliptic curve cryptosystems, and it is at most 50% slower than the equivalent system over finite fields of characteristic two. After that, these elliptic curves were extensively studied by many papers [3, 4, 12, 27]. The literature [29] proposed point addition and point doubling calculation formulas that omit the calculation of the Y-coordinate, and remove the inverse operation, thereby improving the Montgomery algorithm over GF(3^m). The literature [3] improved further point addition, doubling and tripling operation over GF(3^m). In 2013, Gu *et al.* [4] gave the reason why the Montgomery ladder algorithm performs worse over ternary fields than binary fields. In 2015, Zhou *et al.* [25] deduced a formula of calculating 3^k P directly under the affine coordinates. Yu *et al.* [30] optimized Projective Montgomery Algorithm over the finite field with characteristic of 3 by using co-Z tricks, and the Y coordinate is not calculated in the middle of the loop. Robert and Negre [1] first presented thirding point formula together with our third-and-add and parallel approaches for scalar multiplication.

Our contributions in this paper are divided three levels:

- We review researches about scalar multiplication over GF(2^m), GF(p) and GF(3^m) in recent years, and related theory about ECC.
- Different from original Montgomery ladder algorithm, an improved ternary Montgomery ladder algorithm over GF(3^m) is proposed. Furthermore, in order to increase the speed of scalar multiplication, we develop composite operation formulas in the underlying field.
- The proposed algorithm can be applied in elliptic curve cryptography. This algorithm has many advantages over the existing ones, and it has better performance and higher security naturally.

The remainder of this paper is organized as follows. The next section reviews the necessary background for arithmetic on elliptic curves over GF(3^m), ordinary ternary form of k, and related scalar multiplication. In Section 3, we present the improved ternary Montgomery ladder algorithm. Additionally, we derive composite operation formulas which are $2P_1 + P_2$ and $3P_1$. In Section 4, we give some comparison with other algorithms under different coordinate system. Finally, in Section 5, we draw some concluding remarks.

2 Background

2.1 Elliptic Curve Cryptography

An elliptic curve E over a finite field K is defined by the equation

$$y^2 + a_1xy + a_3y = x^3 + a_2x^2 + a_4x + a_6. \quad (1)$$

where $a_1, a_2, a_3, a_4, a_6 \in K$, and$\Delta \neq 0$, the Δ is discriminant of E.

When the characteristic of K is equal to 3, we use the non-supersingular form of an elliptic curve given for $a \neq 0$ by

$$y^2 = x^3 + ax^2 + b. \quad (2)$$

where $a, b \in K$, and $\Delta = -a^3b \neq 0$.

$P_1 = (x_1, y_1) \neq O$ and $P_2 = (x_2, y_2) \neq O$ is two different points on the elliptic curve, then the sum of them is $P_3 = (x_3, y_3)$ computed by

$$x_3 = \lambda^2 - x_1 - x_2 - a, y_3 = \lambda(x_1 - x_3) - y_1, \quad (3)$$

where $\lambda = \dfrac{y_2 - y_1}{x_2 - x_1}$.

The doubling of the point $P_1 = (x_1, y_1) \neq$ O is the point $P_3 = (x_3, y_3)$ given by

$$x_3 = \lambda^2 + x_1 - a, y_3 = \lambda(x_1 - x_3) - y_1, \quad (4)$$

where $\lambda = \dfrac{ax_1}{y_1}$.

Both doubling and addition formulae require $1I+2M+1S$, where I, M, S is the cost of a field inverse, multiplication and squaring, respectively.

2.2 Elliptic Curve over Fields of Characteristic Three

For elliptic curves over fields of characteristic three, we know some basic facts [27]: polynomial g(z), as an element in the field $F_{3^m} = F_3[z]/M(z)$, the cubing rule can be shown as follows:

$$g(z)^3 = g(z^3) \pmod{\text{M(z)}}.$$

The square uses the same algorithm as multiplication, so it is regarded as equivalent to multiplication. Multiplying and squaring field elements are similar in complexity and are not too expensive, but more costly than cubing. The computing speed of cubing is at least ten times faster than that of multiplication or squaring. Since the characteristic of the elliptic curve is 3, so we can simplify cubic computation with Frobenius self-homomorphism [6], and computational overhead of cubic can be negligible, compared to other operations. Addition and subtraction can be ignored as well [2]. Although, the cubing in ternary fields can be implemented by previous efficient methods, the original Montgomery ladder algorithm do not make full use of cubing [4]. In Section 3, we propose an improved ternary Montgomery ladder algorithm. Here, the ternary expression is useful to reduce the length of the scalar representation.

A curve of the form $y^2 = x^3 + ax^2 + b$ is used in this paper, and it has a point of order three and the order of the group is divisible by three. It needs to meet some requirements, for example, there is group order $3 \times p$, where p is a prime, and it can be performed in the subgroup of size p.

2.3 Ordinary Ternary Form

According to the traditional division algorithm, An arbitrary positive integer k is expressed as $k=(k_{n-1},\cdots,k_1,k_0)_3$, where k_{n-1}=1 or 2, and $k_i \in \{0,1,2\}, i=0,1,\cdots,n-2$.

Algorithm 1 Ordinary Ternary Form

Input: A positive integer k
Output: $k=(k_{n-1},\cdots,k_1,k_0)_3$, with k_{n-1}=1 or 2 and $k_i \in \{0,1,2\}$.
$i \leftarrow 0$
while $k>0$ **do**
  if $k \mod 3 = 2$ **then**
    $k_i \leftarrow 2$
    $k = \lfloor k/3 \rfloor$
  end if
  if $k \mod 3 = 1$ **then**
    $k_i \leftarrow 1$
    $k = \lfloor k/3 \rfloor$
  end if
  if $k \mod 3 = 0$ **then**
    $k_i \leftarrow 0$
    $k = k/3$
  end if
  $i \leftarrow i+1$
end while
Output k.
End

Algorithm 1 correctness:

1) The result of k mod 3 in this algorithm is only 2, 1 and 0, and the ratio of 2, 1, and 0 is 1/3 [13], on average.

2) A branch is always performed in the loop, so k will certainly continue to decrease after $k \leftarrow k/3$, the program ends the loop when the final result of k is 0. Therefore,any positive integer k will certainly be turned into an ordinary ternary string, after the circular execution of Algorithm 1.

Example 1. A positive integer $k = 520$

$i \leftarrow 0$
$k_0 \leftarrow 1, k = 173, i \leftarrow 1$
$k_1 \leftarrow 2, k = 57, i \leftarrow 2$
$k_2 \leftarrow 0, k = 19, i \leftarrow 3$
$k_3 \leftarrow 1, k = 6, i \leftarrow 4$
$k_4 \leftarrow 0, k = 2, i \leftarrow 5$
$k_5 \leftarrow 2, k = 0, i \leftarrow 6$
Output $k = \{2,0,1,0,2,1\}$

2.4 Scalar Multiplication

2.4.1 Ordinary Ternary Form Scalar Multiplication

Ordinary ternary form scalar multiplication is expressed as left-to-right form, and Algorithm 2 describes corresponding elliptic curve scalar multiplication.

$$kP = \sum_{i=0}^{n-1} k_i 3^i P = 3(\cdots 3(3k_{n-1}P + k_{n-2}P) + \cdots) + k_0 P.$$

Algorithm 2 Ordinary ternary form scalar multiplication algorithm

Input: $P = (x,y) \in E(GF(2^m))$, and $k = (k_{n-1}, k_{n-2}, \cdots, k_1, k_0)_3$
Output: $Q = kP \in E(GF(2^m))$.
$R_0 \leftarrow O$
for $i = n-1, \cdots, 0$ **do**
  $R_0 = 3R_0$
  $R_1 = R_0 + P$
  $R_2 = R_0 + 2P$
  $R_0 = R_{k_i}$
end for
Return $Q = R_0$
End

The algorithm requires (n)-time triple, (n)-time double and 2(n)-time addition. Each loop performs the same point operation whatever the key bit is, so attacker can not guess the value of scalar k from power trace of point multiplication, *i.e.*, Algorithm 2 can resist SPA.

2.4.2 A Left-to-Right Montgomery Ladder

The well-known Montgomery ladder for speeding up the scalar multiplication, which was initially proposed and utilized for Montgomery form elliptic curves [17], can be adapted to Weierstrass form curves. Algorithm 3 describes the classical left-to-right Montgomery ladder approach for point multiplication [21].

The algorithm requires (n-1)-time double and (n-1)-time addition. Each loop performs one point doubling and one point addition operations, so this algorithm prevents SPA. Furthermore, given a base point P, and there is no change in the difference between the input points R_0 and R_1, *i.e.*, $R_1 - R_0 = P$. This algorithm can also be applied to elliptic curve over GF(3^m) [30].

3 Proposed Algorithm

In this part, we propose a new efficient scalar multiplication algorithm based on the Montgomery ladder algorithm. The scalar k is represented as a ternary form, and the new algorithm is extended to elliptic curves over GF(3^m).

Algorithm 3 Left-To-Right Montgomery Ladder Algorithm

Input: $P = (x, y) \in E(GF(2^m))$, and $k = (1, k_{n-2}, \cdots, k_1, k_0)_2$
Output: $Q = kP \in E(GF(2^m))$.
$R_0 = P$; $R_1 = 2P$
for $i = n-2, \cdots, 0$ **do**
  if $k_i = 1$ **then**
    $R_0 = R_0 + R_1$; $R_1 = 2R_1$
  end if
  if $k_i = 0$ **then**
    $R_1 = R_0 + R_1$; $R_0 = 2R_0$
  end if
end for
Return $Q = R_0$
End

3.1 The Improved Ternary Montgomery Ladder Algorithm

Given a positive integer k, and it is expressed as ternary form $k = 3^{n-1}k_{n-1} + \cdots + 3k_1 + k_0$, where k_{n-1}=1 or 2.

Definition 1. *The value of scalar multiplication is stored in R_0. We define $R_0^{(i)} = (\sum_{j=1}^{i} \mathrm{k}_{n-j} 3^{i-j})P$ as the value of R_0 at the end of the (i-1)-round loop in the algorithm, where $1 \le i \le n$, $j \le i$ and $R_0^{(i)}$ is a point on the elliptic curve. Especially, for $i = 1$, $R_0^{(1)} = (\sum_{j=1}^{1} \mathrm{k}_{n-1} 3^0)P = k_{n-1}P$ is an initial value in the algorithm.*

Similarly, $R_1^{(i)}$ is defined as $R_1^{(i)} = (\sum_{j=1}^{i} k_{n-j} 3^{i-j})P$, and it also holds $R_1^{(i)} = R_0^{(i)} + P$. Then, $R_0^{(i+1)}$and $R_1^{(i+1)}$ are computed by using $R_0^{(i)}$ and $R_1^{(i)}$, and they depend on the value of k_{n-i-1}, as follows:

$$\begin{cases} \text{if } k_{n-i-1} = 0, & R_0^{(i+1)} = 3R_0^{(i)}, \\ & R_1^{(i+1)} = 2R_0^{(i)} + R_1^{(i)} \\ \text{if } k_{n-i-1} = 1, & R_0^{(i+1)} = 2R_0^{(i)} + R_1^{(i)}, \\ & R_1^{(i+1)} = 2R_1^{(i)} + R_0^{(i)} \\ \text{if } k_{n-i-1} = 2, & R_0^{(i+1)} = 2R_1^{(i)} + R_0^{(i)}, \\ & R_1^{(i+1)} = 3R_1^{(i)} \end{cases} \quad (5)$$

Therefore, in the calculation of $R_0^{(i+1)}$ and $R_1^{(i+1)}$, two composite operations $2P_1 + P_2$ and $3P_1$ are involved, where P_1 and P_2 are $R_0^{(i)}$ or $R_1^{(i)}$. $R_1^{(i)} - R_0^{(i)} = P$ is still valid. Algorithm 4 describes the improved ternary Montgomery ladder algorithm over finite fields of characteristic three, and in the following, the algorithm is verified by giving an example.

Notice that at each iteration of Algorithm 4, the variable R_0 is updated as

$$R_0 = \begin{cases} 3R_0, & \text{if } k_i = 0 \\ 2R_0 + R_1, & \text{if } k_i = 1 \\ 2R_1 + R_0, & \text{if } k_i = 2 \end{cases} \quad (6)$$

Algorithm 4 The Improved Ternary Montgomery Ladder Algorithm over GF(3^m)

Input: $P = (x, y) \in E(GF(3^m))$, and $k = (k_{n-1}, k_{n-2}, \cdots, k_1, k_0)_3$, where $k_{n-1} = 1$ or 2
Output: $Q = kP \in E(GF(3^m))$.
$R_0 = k_{n-1}P$, $R_1 = (k_{n-1} + 1)P$
for $i = n-2, \cdots, 0$ **do**
  if $k_i = 0$ **then**
    $R_2 = 3R_0$, $R_1 = 2R_0 + R_1$
  end if
  if $k_i = 1$ **then**
    $R_2 = 2R_0 + R_1$, $R_1 = 2R_1 + R_0$
  end if
  if $k_i = 2$ **then**
    $R_2 = 2R_1 + R_0$, $R_1 = 3R_1$
  end if
  $R_0 = R_2$
end for
Return $Q = R_0$
End

Example 2. $k = 520$, k ={2,0,1,0,2,1}
$R_0 = 2P, R_1 = 3P$
$k = 0, R_0 = 3R_0 = 6P, R_1 = 2R_0 + R_1 = 7P$
$k = 1, R_0 = 2R_0 + R_1 = 19P, R_1 = 2R_1 + 2R_0 = 20P$
$k = 0, R_0 = 3R_0 = 57P, R_1 = 2R_0 + R_1 = 58P$
$k = 2, R_0 = 2R_1 + R_0 = 173P, R_1 = 3R_1 = 174P$
$k = 1, R_0 = 2R_0 + R_1 = 520P, R_1 = 2R_1 + 2R_0 = 521P$
Output $Q = 520P$

and the variable R_1 is updated as

$$R_1 = \begin{cases} 2R_0 + R_1, & \text{if } k_i = 0 \\ 2R_1 + R_0, & \text{if } k_i = 1 \\ 3R_1, & \text{if } k_i = 2 \end{cases} \quad (7)$$

The new proposed algorithm preserves the advantages of the original Montgomery ladder algorithm, the difference between input point R_0 and R_1 is not changed, *i.e.*, $R_1 - R_0 = P$.

3.2 Composite Operation

Computing $2P_1 + P_2$. Let O is the identity element on the elliptic curve, which is considered as a point at infinity.

Now given two points $P_1 = (x_1, y_1)$ and $P_2 = (x_2, y_2)$ in E\{O} with $x_1 \neq x_2$, their sum is the point $R = P_1 + P_2 = (x_3, y_3)$ and is given by using Equation (3)

$$x_3 = \mu_1^2 - a - x_1 - x_2, y_3 = \mu_1(x_1 - x_3) - y_1, \quad (8)$$

where $\mu_1 = \dfrac{y_2 - y_1}{x_2 - x_1}$.

R is added to P_1 to get point $S = 2P_1 + P_2 = (x_4, y_4)$, and coordinates of S is given by

$$x_4 = \mu_2^2 - a - x_1 - x_3, y_4 = (x_1 - x_4)\mu_2 - y_1, \quad (9)$$

where $\mu_2 = \dfrac{y_3 - y_1}{x_3 - x_1}$.

The calculation of y_3 can be omitted by deform μ_2 as

$$\mu_2 = -\mu_1 - \frac{2y_1}{x_3 - x_1} = -\frac{y_1}{x_1 - x_3} - \mu_1.$$

x_4 can be also computed as

$$x_4 = \mu_2^2 - a - x_1 - x_3 = (\mu_2 - \mu_1)(\mu_2 + \mu_1) + x_2.$$

In addition, letting $h := (x_2 - x_1)^2(2x_1 + x_2) - (y_2 - y_1)^2 + a(x_2 - x_1)^2$,it follows that $h = (x_2 - x_1)^2(x_1 - x_3)$. Defining $H := h(x_2 - x_1)$ and $I := H^{-1}$,we get

$$\frac{1}{x_2 - x_1} = hI \text{ and } \frac{1}{\mathrm{x_1 - x_3}} = (\mathrm{x_2 - x_1})^3\mathrm{I}.$$

Therefore, there is no x_3 is used when computing $2P_1 + P_2$. In Algorithm 5, the computation of h, H, I, μ_1 and μ_2 requires 1 inversion, 2 squarings, 1 cubing and 8 multiplications. Similarly, Algorithm 6 is point tripling algorithm.

Algorithm 5 Double-and-Add Algorithm for Elliptic Curve over GF(3^m)

Input: $P_1 = (x_1, y_1) \neq \mathrm{O}$, and $P_2 = (x_2, y_2) \neq \mathrm{O}$
Output: $S = 2P_1 + P_2$.
if $x_1 = x_2$ **then**
  if $y_1 = y_2$ **then**
    return $3P_1$
  end if
  if $y_1 \neq y_2$ **then**
    return P_1
  end if
  $X \leftarrow (x_2 - x_1)^2; Y \leftarrow (y_2 - y_1)^2$
  $h \leftarrow X(2x_1 + x_2) - Y + aX$
  if h=0 **then**
    return O
  end if
  $H \leftarrow h(x_2 - x_1); I \leftarrow H^{-1}$
  $\mu_1 \leftarrow hI(y_2 - y_1)$
  $\mu_2 \leftarrow -y_1X(x_2 - x_1)I - \mu_1$
  $x_4 \leftarrow (\mu_2 - \mu_1)(\mu_2 + \mu_1) + x_2$
  $y_4 \leftarrow (x_1 - x_4)\mu_2 - y_1$
end if
Return (x_4, y_4)
End

4 Analysis of Algorithm

In this part, we first analyze the security of the new algorithm. Then, in order to make the efficiency analysis more accurate, we compare the new algorithm with previous algorithms over GF(3^m) in different coordinate systems. And some practical results are listed in the table below.

Algorithm 6 Tripling Algorithm for Elliptic Curve over GF(3^m)

Input: $P_1 = (x_1, y_1) \neq \mathrm{O}$
Output: $S = 3P_1$.
if $y_1 = 0$ **then**
  return P_1
end if
$A \leftarrow ax_1; B \leftarrow x_1^3 + b$
$C_0 \leftarrow 1; C_1 \leftarrow a(x_1^3 + b) = aB$
$D \leftarrow y_1^3$
$E \leftarrow B^3 - bA^3C_0^2$
$F \leftarrow D^3 - aDC_1^2$
$x_3 \leftarrow \frac{E}{C_1^2}$
$y_3 \leftarrow \frac{F}{C_1^3}$
Return (x_3, y_3)
End

4.1 Security Analysis

The basic idea of power analysis attack is to obtain its key by analyzing the energy consumption during the operation of the cryptographic device. Power analysis attacks include Simple Power Analysis (SPA) and Differential Power Analysis (DPA). SPA [10,14,23] is a technique that can directly analyze the power consumption information, which is collected during the execution of the encryption algorithm. It can retrieve its key through a single leakage trace.

In this paper, the security is analyzed by taking $I/M = 8.75, C/M = 1.37, S/M = 1$ [27,29] for the cost of inverse, cubing and squaring operation. For GF(3^m), the cost of computing $2P_1 + P_2$ is $1I + 2S + C + 8M \approx 1I + 11M$. And the cost of computing $3P_1$ is $1I + 2S + 6C + 3M \approx 1I + 11M$. Because energy consumption of each iteration is the same roughly whether $k_i = 0, 1,$ or 2, so side channel profile of $2P_1 + P_2$ and $3P_1$ can not be distinguished. Our algorithm is based on the original Montgomery ladder algorithm, so the this algorithm is able to resist SPA attacks. Furthermore, this algorithm can resist the DPA by randomizing the scalar.

4.2 Efficiency Analysis

Before we analyze the efficiency, we define β as a dynamic ratio of inverse and multiplication [13]:

$$\frac{I}{M} = \beta. \tag{10}$$

In Algorithm 4, there are operation included tripling and double-and-add or two double-and-add at each loop, which depends on the value of the scalar k. The cost of tripling and double-and-add is $2I + 4S + 7C + 11M$, the cost of two double-and-double is $2I + 4S + 2C + 16M$, so the average cost of each iteration, which is tripling and double-and-add or two double-and-add, is $2I + \frac{38}{3}M + \frac{16}{3}C + 4S$.

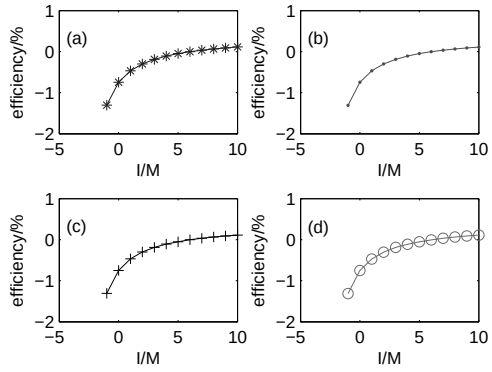


Figure 1: The comparison between Algorithm 3 and Algorithm 4 in the (a) m=101, (b) m=122, (c) m=162, (d) m=379

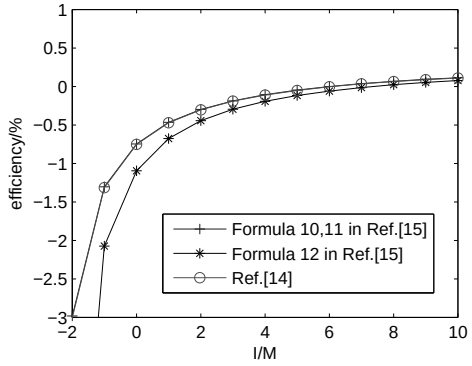


Figure 2: The comparison of algorithms with the scalar of the equivalent bits

4.2.1 Affine Coordinate

The proposed Algorithm 4 is compared with previous algorithm in affine coordinate system. An analysis of the costs of different scalar multiplication is shown in Table 1.

Given an integer k, the ternary (3-adic) expansion of k is shorter than the binary (2-adic) expansion. Suppose k is an n bit number, and $n = \lceil \log_2 k \rceil$, the length of the ternary expansion of k is m, and $m = \lceil \log_3 k \rceil$, therefore, $n = m \log_2 3 \approx 1.584m$, *i.e.*, 101-ternary is equivalent to 160-binary [27], 122, 162, 379-ternary is equivalent to 192, 256, 600-binary, respectively.

For example, comparing Algorithm 3 using formula 10, 11 in [29] and our algorithm, we can conclude that the efficiency of the new algorithm is recorded in the following formula:

$$\varepsilon = 1 - \frac{(2m-2)\beta + (\frac{50m}{3} - \frac{50}{3})}{(2mlog_2 3 - 2)\beta + (6mlog_2 3 - 6)}. \qquad (11)$$

Figure 1 (a)-(d) show that the comparison between the original Montgomery ladder algorithm, *i.e.*, Algorithm 3, and the improved ternary Montgomery ladder algorithm, *i.e.*, Algorithm 4, in different data bits. Our new algorithm can improve the computational efficiency of scalar multiplication with the increase of β. Compared with Algorithm 3 using formula 10, 11 in [29], the average efficiency of the new algorithm can be increased by 6.6% and 11.3% for different data bits, respectively, when β is equal to 8 and 10.

Figure 2 shows the comparison of algorithms with the scalar of the equivalent bits. From the graph of the variation of improved efficiency with the ratio of inverse and multiplication, we can conclude that the larger the ratio of I and M, the slower the rate of increasing in efficiency, and Table 1 shows that the efficiency of the new Montgomery ladder algorithm is increased by 7.8% and 11.0%, compared to Algorithm 3 using formula 12 in [29] and [4], when β is equal to 10.

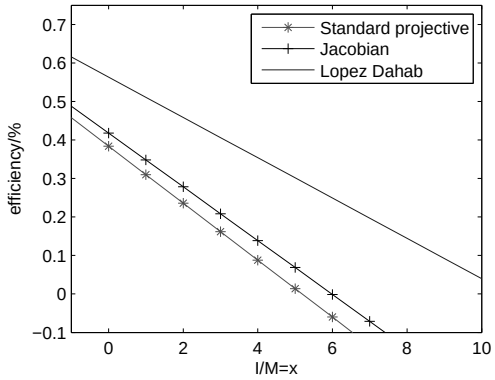


Figure 3: The comparison of algorithms in different coordinates in [27]

4.2.2 Projective Coordinate

As is shown in Table 2, the proposed Algorithm 4 is compared with previous algorithm, such as [27] and [30], in projective coordinate system. The peculiarity of this article is the dynamic ratio β, resulting in a dynamic percentage. In this paper, we take a list of the efficiency at special points.

Figure 3 shows the comparison of algorithms in different projective coordinates in [27], we can draw that the efficiency decreases, as the ratio increases. For algorithm in Lopez Dahab coordinate, it has a slow rate of decline, compared with the other two projective coordinates. Form Table 2, we draw that if we set the maximum value of the ratio as 10, the efficiency is improved by 4.0%.

Figure 4 shows the comparison of algorithms in different projective coordinates in [30]. Compared with the other two previous algorithm, the algorithm in Co-Z projective has higher efficiency. In addition, in Table 2, compared our algorithm, previous algorithm is proposed by 5.0% in standard projective, when β is equal to 3, and 7.2% in scaled projective [3], when β is equal to 2, and 4.9% in Co-Z projective, when β is equal to 1.5, respectively.

Table 1: Timing costs of different algorithm in affine coordinate system

Algorithm	**Total costs**($n = \lceil \log_2 \boldsymbol{k} \rceil, m = \lceil \log_3 \boldsymbol{k} \rceil$)	**Total costs**($\#I + \#M$)
Formula 10,11 in [29]	$2(n-1)I + 4(n-1)M + 2(n-1)S$	$(2n-2)I + (6n-6)M$
Formula 12 in [29]	$2(n-1)I + 3(n-1)M + 2(n-1)C + 2(n-1)S$	$(2n-2)I + (5n-5)M$
[4]	$(2n-3)I + (3n-5)M + (3n-5)S$	$(2n-3)I + (6n-10)M$
Ours	$2(m-1)I + \frac{38}{3}(m-1)M + \frac{16}{3}(m-1)C + 4(m-1)S$	$(2m-2)I + \frac{50}{3}(m-1)M$

Table 2: Timing costs of different algorithm in projective coordinate system

Algorithm	**Coordinate**	**Each iteration's costs**	**Total costs**($\#M$)	$\frac{I}{M} = \beta$	ε
Ref. [27](binary)	Standard projective	$15M + 2S + 4C$	$17M$	4	8.8%
	Jacobian	$13M + 5S + 5C$	$18M$	5	6.9%
	Lopez Dahab	$17M + 7S + 2C$	$24M$	8	17.9%
				10	4.0%
Ref. [30](binary)	Standard projective	$13M + 2S + 2C$	$15M$	3	5.0%
	Scaled projective [3]	$14M + 3C$	$14M$	2	7.2%
	Co-Z projective	$10M + 3S + C$	$13M$	1.5	4.9%
Ours(ternary)	Affine	$2I + \frac{38M}{3} + \frac{16C}{3} + 4S$	$2I + \frac{50M}{3}$		

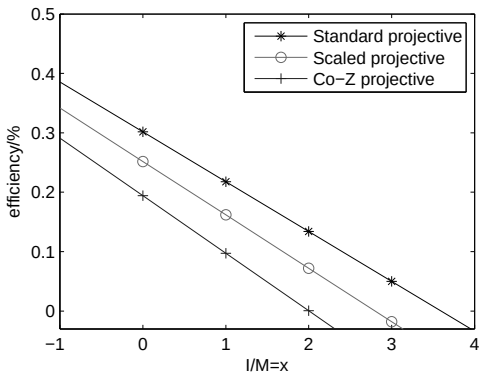


Figure 4: The comparison of algorithms in different coordinates in [30]

In the actual experimental environment, I/M is different and relatively large. Table 2 shows that the efficiency of new algorithm varies from 4% to 17.9%, compared with previous algorithm in projective coordinate system.

5 Conclusions

In this paper, we proposed an improved Montgomery ladder algorithm over GF(3^m), and the scalar was expressed in ternary form. In addition, we derived composite operation formulas $2P_1 + P_2$ and $3P_1$ with a lower computational cost. Based on the original Montgomery ladder algorithm, it is able to resist SPA. In analyzing efficiency, the difference with the past is that the ratio of I to M, which is set to a dynamic value. Correspondingly, increased efficiency is also dynamic, ranging from 7.8% to 11.3% in affine coordinate, and from 4% to 17.9% in projective coordinate. Further work may include designing a new scalar multiplication algorithm over GF(3^m), and it makes cubing operation faster.

References

[1] N. Christophe and J. M. Robert, "New parallel approaches for scalar multiplication in elliptic curve over fields of small characteristic," *IEEE Transactions on Computers*, vol. 64, no. 10, pp. 2875–2890, 2015.

[2] M. Ciet, M. Joye, K. Lauter, and P. L. Montgomery, "Trading inversions for multiplications in elliptic curve cryptography," *Designs, Codes and Cryptography*, vol. 39, no. 2, pp. 189–206, 2006.

[3] R. R. Farashahi, H. F. Wu, and C. A. Zhao, "Efficient arithmetic on elliptic curves over fields of characteristic three," in *International Conference on Selected Areas in Cryptography*, pp. 135–148, Aug. 2012.

[4] H. H. Gu, W. L. Xie, and Ray C. C. Cheung, "Analysis the montgomery ladder algorithm for elliptic curves over ternary fields," in *International Conference on Information and Network Security (ICINS'13)*, pp. 1–5, Nov. 2013.

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

[6] D. Hankerson, A. Menezes, and S. Vanstone, *Guide to Elliptic Curve Cryptography*, Springer-Verlag New York, 2004. (`http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.394.3037&rep=rep1&type=pdf`)

[7] G. F. Hou and Z. J. Wang, "A robust and efficient remote authentication scheme from elliptic curve cryptosystem," *International Journal of Network Security*, vol. 19, no. 6, pp. 904–911, 2017.
[8] C. Kakali, D. Asok, and Daya Gupta, "Mutual authentication protocol using hyperelliptic curve cryptosystem in constrained devices," *International Journal of Network Security*, vol. 15, no. 1, pp. 9–15, 2013.
[9] N. Koblitz, "Elliptic curve cryptosystems," *Mathematics of Computation*, vol. 48, no. 177, pp. 203–209, 1987.
[10] P. Kocher, J. Jaffe, and B. Jun, "Differential power analysis," *Lecture Notes in Computing Science Springer-Verlag*, vol. 1666, pp. 388–397, 1999.
[11] K. Koyama and Y. Tsureoka, "Speeding up elliptic curve cryptosystems by using a signed binary windows method," *Advances in Cryptology (CRYPTO'92)*, pp. 345–357, 1992.
[12] H. K. Kwang, I. K. So, and S. C. Ju, *New Fast Algorithms for Arithmetic on Elliptic Curves Over Finite Fields of Characteristic Three*, May 2007. (`http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.81.1113`)
[13] L. Li, "Research on the ternary algorithm in the elliptic curve operations," *Journal of Network Safety Technology and Application (in Chinese)*, no. 11, pp. 94–96, 2015.
[14] S. G. Liu, H. T. Yao, and X. A. Wang, "Spa resistant scalar multiplication based on addition and tripling indistinguishable on elliptic curve cryptosystem," in *10th International Conference on P2P, Parallel, Grid, Cloud and Internet Computing (3PGCIC'15)*, pp. 785–790, Nov. 2015.
[15] J. Lopez and R. Dahab, "Fast multiplication on elliptic curves over gf(2m) without precomputation," in *First International Workshop on Cryptographic Hardware and Embedded Systems (CHES'99)*, pp. 316–327, Aug. 1999.
[16] V. S. Miller, "Use of elliptic curves in cryptography," *Lecture Notes in Computer Science Springer-Verlag*, vol. 218, no. 1, pp. 417–426, 1986.
[17] P. L. Montgomery, "Speeding the pollard and elliptic curve methods of factorization," *Mathematics of Computation*, vol. 48, no. 177, pp. 243–264, 1987.
[18] J. Moon, D. Lee, and J. Jung, "Improvement of efficient and secure smart card based password authentication scheme," *International Journal of Network Security*, vol. 19, no. 6, pp. 1053–1061, 2017.
[19] C. Negre, "Scalar multiplication on elliptic curves defined over fields of small odd characteristic," in *Proceedings of the 6th International Conference on Cryptology (INDOCRYPT'05)*, pp. 389–402, Dec. 2005.
[20] K. Okeya and T. Takagi, "The width-w naf method provides small memory and fast elliptic saclar multiplications secure against side channel attacks," *Rsa Conference on the Cryptographers*, pp. 328–343, 2003.
[21] T. Oliveira, J. López, and R. H. Francisco, "The montgomery ladder on binary elliptic curves," *Journal of Cryptographic Engineering*, no. 5, pp. 1–18, 2017.
[22] Q. Qian, Y. L. Jia, and R. Zhang, "A lightweight rfid security protocol based on elliptic curve cryptography," *International Journal of Network Security*, vol. 18, no. 2, pp. 354–361, 2016.
[23] Reddy and E. Kesavulu, "Elliptic curve cryptosystems and side-channel attacks," *International Journal of Network Security*, vol. 12, no. 3, pp. 151–158, 2011.
[24] T. Satoh, "The canonical lift of an ordinary elliptic curve over a finite field and its point counting," *Journal of the Ramanujan Mathematical Society*, vol. 15, no. 4, pp. 247–270, 2000.
[25] S. F. Shen and M. Zhou, "Research on fast algorithms for scalar multiplication of elliptic curve cryptography over gf(3n)," *Advances in Applied Mathematics (in Chinese)*, vol. 4, no. 4, pp. 390–399, 2015.
[26] C. S. Sin, *Regular Ternary Algorithm for Scalar Multiplication on Elliptic Curves Over Finite Fields of Characteristic Three*, July 2012. (`https://eprint.iacr.org/2012/390.pdf`)
[27] N. P. Smart and E. J. Westwood, "Point multiplication on ordinary elliptic curves over fields of characteristic three," *Applicable Algebra in Engineering, Communication and Computing*, vol. 13, no. 6, pp. 485–497, 2003.
[28] S. F. Tzeng, M. S. Hwang, "Digital signature with message recovery and its variants based on elliptic curve discrete logarithm problem", *Computer Standards & Interfaces*, vol. 26, no. 2, pp. 61–71, Mar. 2004.
[29] H. Wang, B. Li, and W. Yu, "Montgomery algorithm on elliptic curves over finite fields of character three," *Journal on Communications (in Chinese)*, vol. 29, no. 10, pp. 25–29, 2008.
[30] W. Yu, B. LI, K. P. Wang, W. X. Li, and S. Tian, "Co-z montgomery algorithm on elliptic curves over finite fields of characteristic three," *Chinese Journal of Computers (in Chinese)*, vol. 40, no. 5, pp. 1121–1133, 2017.

Biography

Shuang-Gen Liu is PhD, associate professor. His research interests are information security and cryptology.

Rong-Rong Wang is a graduate student of Xi'an University of posts and telecommunications. She is mainly engaged in the research of elliptic curve cryptosystem.

Yun-Qi Li is a undergraduate student in information security in Xi'an University of posts and telecommunications. Her research interest is cryptography.

Can-Liang Zhai is a undergraduate student in information security in Xi'an University of posts and telecommunications. His research interest is cryptography.

A Fast Recovery Method for Single Disk Failure Based on EVENODD

Feng Xiao, Di Fan, and Dan Tang
(Corresponding author: Dan Tang)

School of Software Engineering, Chengdu University of Information Technology
No. 24, Xue-Fu Road, Chengdu 610225, China
(Email: 398879332@qq.com)

(Received Nov. 23, 2017; Revised and Accepted Apr. 28, 2018; First Online Jan. 12, 2019)

Abstract

In an array storage system based on EVENODD, the traditional single disk failure recovery requires reading all the remaining data. Even if the hybrid recovery method is used, the theoretical limit of the data reading cost can be reduced by only 25%.To further reduce the read overhead for single disk failure recovery, improve the reliability of storage system, this paper proposes an improved method for EVENODD single disk failure recovery. This method is to reduce the amount of data that needed to read during recovery by adding a local redundant.The experimental results show that the improved method has significantly improved the recovery time comparing with the EVENODD.

Keywords: EVENODD; Reading Overhead; Single Disk Failure

1 Introduction

In recent years, with the rapid development of society and technology, the amount of data in the world is increasing at an explosive rate. As a result, the scale of storage systems is also growing accordingly. As the scale increases, the stability of the storage system is also facing many great challenges. The main reason is that while the number of disks is increasing, the probability of data loss caused by disk failures is also increasing. In order to protect the data, it is necessary to recover the data even after the disk failed. Therefore, the time of data recovery and the amount of the data that needed to be read during the recovery process become the key to the stability of the system.

Array codes is a kind of erasure code which is widely applied. Its advantage is the calculation is XOR so the encoding and decoding speed are pretty fast. The typical array codes include EVENODD [5], RDP [3], STAR [7], RTP [4] and so on. STAR and RTP are expanded by EVENODD and RDP respectively. However, there are still many problems to be solved. Aiming at the problem of update efficiency, Chen put forward the Inverse Code [2] by constructing low density matrix. Aiming at the problem of data storage reliability and expansibility, Teng proposed the Random Array Code [10]. In order to accelerate the speed of data recovery, Sun proposed a new coding theory named ESRC [9], which is based on local check and global check. In order to solve the problem that the MDS array codes have a length limitation, Huang proposed a new code named Symmetric code [6].

EVENODD code is one of the most commonly used array codes in storage systems which can tolerate two errors. Its principle is to ensure the integrity of the data by adding two redundant disks. EVENODD can recover the original data correctly after any two of the disks failed.

For EVENODD coding, the advantages are that the structure is simple, and the encoding and decoding process are based on XOR operation, so the speed is very fast. However, one of the major defect is that the single disk failure recovery requires too much data to read. All the remaining raw data is need to be read. Considering that the recovery is based on a single check disk, if a local check disk is computed only by some of the data disks, then when the data disk fails, only part of the data disks that generate the local check disk can be recovered. Therefore, a fast recovery method for single disk failure based on EVENODD is proposed.

2 Introduction of EVENODD

2.1 Encoding Process

The encoding process of EVENODD is based on an original data array, and the size of the original data array is $(p-1)*p$, p is a prime number. The data block in the array is assumed to be $a_{i,j}$, where i and j represent row and column coordinates of elements. The coding formulas are shown in Equations (1),(2)and (3):

$$a_{i,p} = \bigoplus_{j=0}^{p-1} a_{i,j} \tag{1}$$

$$S = \bigoplus_{j=1}^{p-1} a_{p-1-j,j} \quad (2)$$

$$a_{i,p+1} = S \oplus \left(\bigoplus_{j=0}^{p-1} a_{<i-j>_p,j} \right). \quad (3)$$

The final coding process is shown in Figure 1.

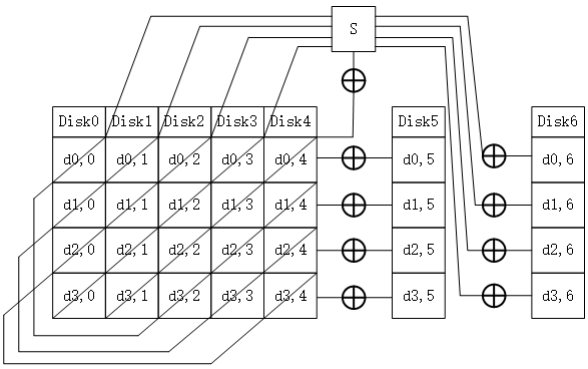


Figure 1: Horizontal redundancy and diagonal redundancy calculation process

Finally, an array of $(p-1)*(p+2)$ with two redundant columns is obtained.

2.2 Single Disk Failure Recovery

The traditional single disk failure recovery method can be divided into two categories according to the position of the failed disk: data disk failure and redundant disk failure.

The method of recovering from data disk failure is to determine the location of the failed disk first, and then read out all the other data blocks in data disks and horizontal redundant disk, and finally run XOR operation through these data, get the lost data disk. The data required to be read during recovery is shown in Figure 2, the "X" indicates that the data block is lost and "O" means the data block need to be read.

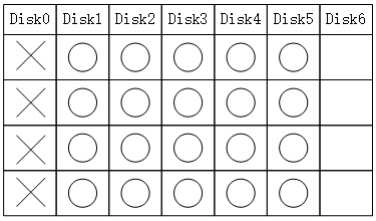


Figure 2: Data to be read from single disk failure recovery

The recovery of the redundant disk failure is exactly the same as the coding process, that is, when the horizontal or diagonal redundant disk fails, it can be rebuilt by using encoding algorithm.

Therefore, for the method of traditional single disk failure recovery, whether the data disk failed or redundant disk failed, the number of data disks required is p.

2.3 Single Disk Failure Hybrid Recovery Algorithm

In order to reduce the recovery read overhead of single disk failure, literature [1] proposes that, for EVENODD when a data disk fails, the horizontal redundancy and diagonal redundancy can be used simultaneously to recover. Because there are some repeated data blocks needed in using different methods to recover, so the recovery read overhead can be reduced by caching these blocks. Theoretical deduction is also carried out in this paper, and it is found that the best recovery method is to recover half of the data using horizontal redundancy, and recover the other half data using diagonal redundancy. And the lower bound of the theoretical recovery read overhead is 3/4 of the total data, which means it can save 25% of the recovery read overhead. Similarly, a hybrid recovery algorithm for RDP single disk failure is proposed in the literature [11]. The lower bound of the read overhead is also 3/4 of the total data.

As it's shown in Figure 3, when the Disk0 fails, the first two data blocks are recovered by horizontal redundant columns, and the latter two data blocks are recovered by diagonal redundancy. Among them the circle indicates that the data blocks used by horizontal redundant columns while square means the data blocks used by diagonal redundant columns, and the data blocks which have both symbols mean they are repeated.

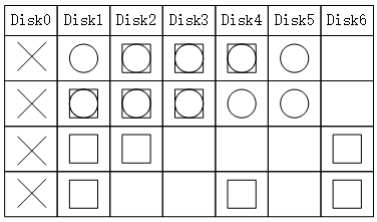


Figure 3: Using hybrid algorithm to recover single disk failure

3 Local Repair Method of EVENODD

In order to reduce the recovery read overhead of EVENODD code in the single disk failure further, this paper has modified this code. The transformation method is called EVENODD local repair method. And the expanded EVENODD is called LREVENODD (Local Repairable EVENODD).

3.1 Encoding Process

Local repairable EVENODD code is an extension of EVENODD. A redundant column x is added on the basis of the original EVENODD with two redundant columns. Redundant column x is obtained by horizontal XOR computation of the previous $\frac{p-1}{2}$data columns. The computation formula is Equation (4):

$$a_{i,p+2} = \bigoplus_{j=0}^{\frac{p-1}{2}} a_{i,j} \tag{4}$$

Encoding process of Local repairable EVENODD code is shown in Figure 4. The same shape of white blocks generate the corresponding black check blocks:

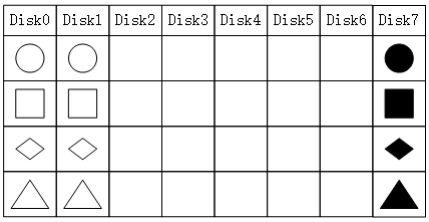


Figure 4: Encoding process of LREVENODD

3.2 Encoding Overhead

For the EVENODD local repair method, encoding process only adds a step in the traditional process of EVENODD encoding. In the calculation of horizontal redundant columns, after XORing the first half of the data, the XOR result as local redundant column and save to disk X. Although this step increases the storage overhead and the encoding time, it does not increase the computational complexity. Moreover, with the increase of prime number, the proportion of increased overhead to the overall coding overhead will continue to decline. Because the value p in the real environment represents the number of disks in the storage cluster, it is generally larger, so the overhead caused by the increased local redundant column x is relatively smaller.

3.3 Correctness Proof

Based on the definition in the upper section, the former $p+2$ column of the Local Repairable EVENODD code is the EVENODD code, so the nature of the former $p+2$ column is no longer discussed at next, and only the impact of the new local redundant column x loss is discussed

3.3.1 Only the X Column is Missing

If only the X column is missing, it means that all the data in the front $p+2$ columns is in good condition, and the X column can be generated by encoding Equation (4). At this point, the data integrity can be guaranteed.

3.3.2 X and Another Column Y Missing

First, suppose that another missing column is a data column, then the horizontal redundant column is intact. So the data column can be recovered by XOR of all the remaining data columns and the horizontal redundant column. The computation formula is Equation (5):

$$a_{i,y} = \bigoplus_{j=0}^{p} a_{i,j} \tag{5}$$

When all the data columns have been rebuilt, the situation is consistent with the loss of the X column, and then the X column can be generated again by encoding.

If another missing column is a redundant column, which means all data columns are in good condition. So at first, the missing horizontal redundant column or diagonal redundant column can be regenerated by EVENODD encoding. If the horizontal redundancy column is lost, the coding formula of recovery is Equation (1). If the missing redundant column is diagonal redundant column, the coding formula of recovery is shown in Equation (2) and Equation (3).

After restoring the horizontal redundant column or diagonal redundant column, only the X column still needs to be recovered. So the X column can be restored by Equation (4).

Through the analysis above, it can be concluded that after adding a local redundant column x, the new array code can still recover any two errors.

3.4 Relative Properties

3.4.1 MDS Property

It can be found that although the extended EVENODD code adds a redundancy, but it can only tolerate any two errors (in some cases it can tolerate three errors) and lose the MDS property. The reason is that the local fault tolerance is in conflict with MDS property. For example, RS codes remove the lost blocks and decode them with the rest of the global blocks, so the locality is poor, but there is good code distance, and has the best fault tolerance. EVENODD and all the MDS codes are like this. The literature [8] pointed out that if adds some local redundant columns, it means that there exists errors(missing columns is not related to these columns) that the local redundant columns can not be solved. Thus it can be seen that the local fault tolerance and minimum code distance are conflicted, which means it is inevitable to improve the local tolerance while sacrificing MDS property.

Furthermore, it is proved that the Local Repairable Codes with additional local redundant columns can be infinitely approximated to MDS codes. Then, for a MDS code with only one redundant column, it can be proved that, when the value p is increased, the LREVENODD code can also be close to the MDS codes.

3.4.2 The Relationship Between New Redundancy and Overhead Reduction

By discussing MDS property, it can be concluded that increasing local redundant columns and ensuring MDS property are conflicted. The paper [12] pointed out that appropriate data redundancy can improve system performance. Therefore, it is necessary to explore the relationship between redundant columns number and reduced overhead to determine how much local redundancy is added to achieve optimal.

In order not to increase the complexity of the encoding process, the redundant columns are selected by sharing the original data columns. The number of local redundant columns is n, and the percentage of saved overhead is cp, then the ratio of new redundancy and overhead is $l = \frac{cp}{n}$. Assuming that the additional local redundancy columns are 1, 2, 3, 4 and 5, the obtained results are shown in Table 1:

Table 1: Relationship between new redundancy and overhead reduction

symbol	value				
cp	50%	67%	75%	80%	83%
n	1	2	3	4	5
l	50%	33%	25%	20%	17%

It can be seen from the table that with the increase of the number of local redundant columns, the saved read overhead is also increasing, but the reduction of the overhead caused by the new redundancy is decreasing, that is, the efficiency is decreasing. And adding redundant columns will bring more storage overhead, and will also influence the efficiency of encoding and decoding. So the choice of adding only a local redundant column can significantly reduce the single disk failure recovery read overhead and does not increase the excessive storage overhead and lead to system performance degradation.

4 Decoding Process

4.1 Single Disk Failure

For traditional EVENODD coding, the recovery of single disk failure (non redundant columns) is done by reading the remaining data columns and the horizontal redundant columns to xor. For the EVENODD local repair method, there are two situations to repair a single disk failure.

First, suppose the data column fails, then it can be divided into two cases, one is the missing data column is in the first half, and the other is in the second half. If the missing column is in the first $\frac{p-1}{2}$ columns, it is necessary to read the first half part of the remaining data column and the new local redundant column x for XOR restoration. The data block needed to recover is shown in Figure 5, the "X" indicates that the data block is lost, "O" means the data blocks which are required in recover.

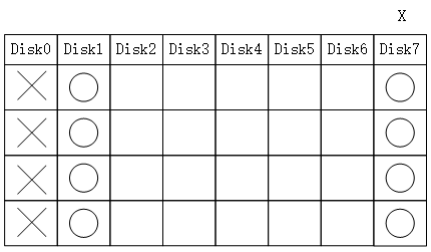


Figure 5: Data need to read when missing column is in first part

If the missing column is in the posterior $\frac{p+1}{2}$ columns, the local redundancy column X and the horizontal redundant column p can be read first, then XOR them, and the result is the XOR of the data column in the second half. Then read the data columns in the second half, and XOR with the previous result, to get the missing data column. The data blocks needed for recovery are shown in Figure 6.

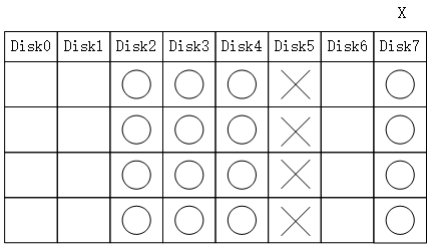


Figure 6: Data need to read when missing column is in second part

It can be concluded that the amount of data needed to recover from single disk failure is $\frac{p-1}{2}$ and $\frac{p+1}{2}$ respectively in two cases. When p is increasing, the reduced read overhead tends to approach 50%.

If the local redundant column fails, it is necessary to read the first $\frac{p-1}{2}$ data columns for restoration, which is completely equivalent to the encoding process. This situation does not exist in the EVENODD codes, and the recovery read overhead remains $\frac{p-1}{2}$. When p is increasing, the proportion of local redundant column fails to all single disk failures will continue to decrease.

In case of horizontal redundant column fails, only the posteriors $\frac{p+1}{2}$ data columns and the local redundant column are needed to recover. Read overhead is only $\frac{p+1}{2}+1$. When p is increasing, it can reduce the data reading overhead approaching to 50%. The data blocks required for the recovery of horizontal redundant column failure are shown in Figure 7.

For the case of diagonal redundant column fails, the EVENODD local repair method is consistent with the traditional EVENODD recovery algorithm, that is, read all

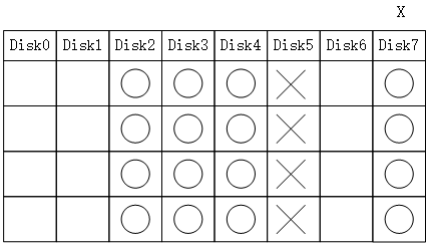


Figure 7: Data need to read when horizontal redundant column is missing

data columns to encode and get the redundant columns, and the read overhead is p. Similarly, the ratio of this failure to all single disk failures will continue to decrease as p continues to increase. Therefore, for single disk failure recovery, assuming that the probability of each disk failure is consistent, the average overhead of the LREVENODD single disk failure recovery is shown in Equation (6):

$$c = \frac{\frac{p-1}{2} * \frac{p-1}{2} + \frac{p+1}{2} * \frac{p+3}{2} + \frac{p+3}{2} + p + \frac{p-1}{2}}{p+3} \tag{6}$$

According to the analysis above, it can be concluded that the single disk failure recovery overhead of LREVENODD approach nearly $\frac{p}{2}$, and Figure 8 is a theoretical comparison between the EVENODD and the LREVENODD single disk failure recovery overhead when p takes different values.

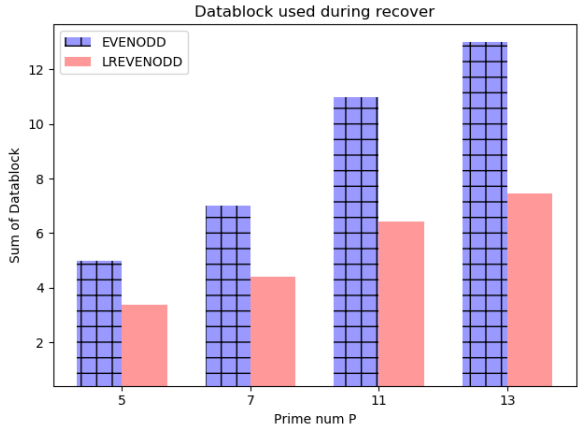


Figure 8: Comparison of single disk failure recovery reading overhead

Compared with the traditional EVENODD recovery algorithm, the proposed method saves 50% of the data read overhead in the single disk failure recovery, while the recovery of redundant columns is almost consistent with the traditional EVENODD read overhead. Because the number of data columns in practical applications is much larger than the number of redundant columns, it can be concluded that the method reduces the data read overhead by nearly 50% for single disk failure recovery.

4.2 Double Disk Failures Recovery

4.2.1 Two Data Column Missing

Like traditional recovery methods, recovery is done by reading the remaining data and using horizontal redundancy and diagonal redundancy when two data columns failed in LREVENODD. Data read overhead is shown as Equation (7):

$$cp = p - 2 + 2 = p. \tag{7}$$

4.2.2 One Data Column and Local Redundant Column Missing

First, by reading the remaining data columns and horizontal redundant column to recover the lost data column, and cache the xor result of the previous $\frac{p-1}{2}$ data columns(don't xor the missing column, but xor it with the cache result when restored it), the final xor result is the missing local redundant column.

As shown in Figure 9, when the second data column and the local redundant column are lost, the remaining intact data columns and horizontal redundant columns are first read to recover the lost second data column. At the same time, the XOR results are saved in the process. After calculating the second column data, xor the result with the second data column, and the local redundant column is obtained.

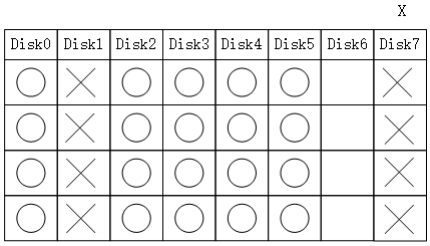


Figure 9: Data need to read when local redundant column and one data column are missing

As shown in Figure 9, the read overhead of recovery in this case is still p columns.

4.2.3 One Data Column and Horizontal or Diagonal Redundant Column Missing

The recovery method in this situation is consistent with the traditional EVENODD recovery method. The reading overhead is p.

4.2.4 Two Redundant Columns Missing

The recovery method in this situation is consistent with the encoding process. So obviously, the reading overhead is p.

In summary, the EVENODD local repair method can greatly reduce the read overhead and improve the performance when the single data disk fails. For other cases,

the EVENODD local repair method does not produce too much overhead, and is basically consistent with the traditional EVENODD.

4.3 Triple Disk Failures Recovery

Although the extended EVENODD loses its MDS property, it can still be successfully restored for most of the three disk failures. Then analysis in which case the three disk failure can be restored when $p = 7$.

The three redundant columns in the extended EVENODD can be represented by the following equations:

$$\begin{aligned} a_{i,p} &= \bigoplus_{j=0}^{p-1} a_{i,j} \quad (i = 0, 1, ..., p-2) \\ a_{j,p+1} &= S \oplus \left(\bigoplus_{j=0}^{p-1} a_{<i-j>_p,j} \right) \quad (i = 0, 1, ..., p-2) \\ a_{i,p+2} &= \bigoplus_{j=0}^{\frac{p-1}{2}} a_{i,j} \quad (i = 0, 1, ..., p-2). \end{aligned}$$

Meanwhile, assume the probability of each disk failure is γ.

4.3.1 Three Data Disks Missing

When three disks are lost and the three disks are all data disks, the probability is shown as Equation (8).

$$\lambda = C_p^3 * \gamma^3 \tag{8}$$

When the lost disks are all data disks, for the horizontal redundant disk, each of the equations contains 3 unknown quantities. For diagonal redundant disk, each equation contains at least 2 unknown quantities. The problem then turns into whether the situation can be translated into two disks lost, i.e. whether a disk exists in the three lost disks can be recovered by the local redundant disk. We discuss the situation according to the position distribution of the three data disks:

First, the three column data disks are in the front $\frac{p-1}{2}$ disks or back $\frac{p+1}{2}$ disks, the probability of such a situation is shown as Equation (9).

$$\lambda = C_{\frac{p-1}{2}}^3 * \gamma^3 + C_{\frac{p+1}{2}}^3 * \gamma^3 \tag{9}$$

First, suppose that three disks are in the previous $\frac{p-1}{2}$ disks, as shown in Figure 10. Assuming each column represents a data disk, for a local redundant disk, each of its equations contains 3 unknown quantities and is not solvable. And if three disks are in posteriors $\frac{p+1}{2}$ disks is also similar, because the disks in the back are not a linear relationship with the local redundant disk, it is impossible to recover the local redundant disk. And only through the other two redundant disks to restore has too many unknown disks, so this situation is not solvable.

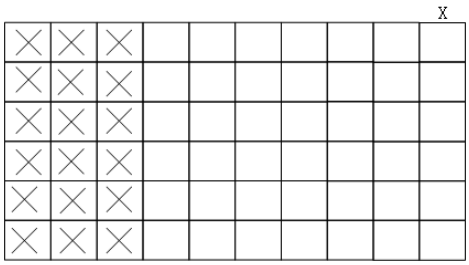


Figure 10: Can not recover from missing three data disks in same part

Assuming there is a disk in the three data disks not in the same half with the other two disks, the probability of this kind of case is shown as Equation (10).

$$\lambda = C_{\frac{p-1}{2}}^1 * C_{\frac{p+1}{2}}^2 * \gamma^3 + C_{\frac{p-1}{2}}^2 * C_{\frac{p+1}{2}}^1 * \gamma^3 \tag{10}$$

First, suppose that there is a lost disk in the previous $\frac{p-1}{2}$ disks and the other two lost disks are in the second half, then there is only one unknown quantity in each equation that constitutes a local redundant disk. So it can be solved. When the first lost disk is restored, the problem is transformed into an ordinary double disk failure, so the situation is solvable.

If there are two disks lost in the previous $\frac{p-1}{2}$ disks and one in the second half. Then the equations of the horizontal redundant disk and the equations that constitute the local redundant disk can be subtracted, and a set of equations about the latter $\frac{p+1}{2}$ disks can be obtained, shown as Equation (11).

$$a_{i,p} - a_{i,p+2} = \bigoplus_{j=\frac{p-1}{2}}^{p-1} a_{i,j} \tag{11}$$

In these equations, each has only one unknown quantity and can be solved. When the missing data is restored, the problem is transformed into an ordinary double disk failure again. So the situation is also solvable.

4.3.2 Two Data Disks and One Redundant Disk Missing

When there are three disks lost, two of them are data disks, and the other one is redundant disk, and the probability of this case is shown as Equation (12).

$$\lambda = C_p^2 * C_3^1 * \gamma^3 \tag{12}$$

This situation can be divided into several types according to the redundant disk type, in which the probability of each case is shown as Equation (13).

$$\lambda = C_p^2 * \gamma^3 \tag{13}$$

First, if the lost disk is a local redundant disk, it is equivalent to the normal EVENODD double disk failure

recovery. After the completion of the restoration, the local redundant disk can be restored by encoding.

If the lost disk is horizontal redundant disk, then it can be classified according to the distribution of the data disks.

When the two disks are in the first half, the property of EVENODD shows that there is always only one unknown $a_{i,j}$ in the equations of the diagonal redundant disk. Through diagonal redundancy can calculate this unknown quantity $a_{i,j}$. And then there is only the unknown quantity $a_{i,k}$ in the same row, and it can be calculated through the local redundancy. And after $a_{i,k}$ is calculated, there is only one unknown quantity on the line and it located which slope is 1. And by repeating these steps, it can solve all the unknowns.

If the two disks are in the second half, because the local redundancy is independent of the latter $\frac{p+1}{2}$ disks, they can only be solved by diagonal redundancy. The previous discussion shows that although the diagonal redundancy can solve an unknown quantity $a_{i,j}$, but the other unknown quantity in the same line depends on other redundancy to solute. While there is no other redundant existence at the same time, so it can not be solved. The probability of this case is shown as Equation (14).

$$\lambda = C^2_{\frac{p+1}{2}} * \gamma^3 \tag{14}$$

Assuming there is a data disks in the previous $\frac{p-1}{2}$ disks while the other is in the second part, the probability of such a situation is shown as Equation (15).

$$\lambda = C^1_{\frac{p-1}{2}} * C^1_{\frac{p+1}{2}} * \gamma^3 \tag{15}$$

There is only one unknown quantity existed in each equation that constitutes a local redundant disk. So it can be solved. When the first lost disk is restored, there is still one data disk and horizontal redundant disk are missing, which can be restored by the traditional RDP double disk recovery method. So this situation is solvable.

If the lost disk is redundant disk, then it can be classified according to the distribution of the data disk.

Assuming the two disks are both in the previous $\frac{p-1}{2}$ disks or in the second half, the probability of this happening is shown as Equation (16).

$$\lambda = \left(C^2_{\frac{p-1}{2}} + C^2_{\frac{p+1}{2}}\right) * \gamma^3 \tag{16}$$

When the two lost data disks are in the first half, even though there are two redundant disks, the two redundant disks are linearly related, so the two missing disks can not be restored. And when the two disks are in the second half, there is only horizontal disk related to the lost disk, so it can not be recovered either.

If one of the two disks is in the first half and the other in the second half, they can be restored. Because the lost disk in the first half can be restored by the local redundant disk, the remaining lost data disk can be restored by the horizontal redundant disk. Finally, the diagonal redundant column is restored by the encoding algorithm.

4.3.3 One Data Disk and Two Redundant Disks Missing

Assuming there are three disks lost, one is a data disk, and the other two are redundant disks, the probability of this case is shown as Equation (17).

$$\lambda = C^1_p * C^2_3 * \gamma^3 \tag{17}$$

In consideration of that only one data disk is lost, if the intact redundant disk is a horizontal redundant disk or a diagonal redundant disk, the data disk can be restored, and then the lost redundant disks can be restored by encoding.

Assuming the remaining redundant disk is local redundant disk, the probability of such a situation is shown as Equation (18).

$$\lambda = C^1_p * \gamma^3 \tag{18}$$

This situation needs to be classified according to the location of the data disk.

If the lost data disk is in the front half, the data disk can be recovered through a local redundant disk, and then other redundant disks are restored by encoding.

And assuming the missing data disk is in the second half, the probability of such a situation is shown as Equation (19).

$$\lambda = C^1_{\frac{p+1}{2}} * \gamma^3 \tag{19}$$

Because the local redundant disk is linear independent of the latter part, and it is impossible to recover the lost data disk only by diagonal redundancy, so the situation is not solvable.

4.3.4 Three Redundant Disks Missing

If the lost three disks are redundant disks, they can be restored directly by encoding again.

From the above discussion, we can get the sum of the probability of the non-solvable cases, which are shown as Equation (20).

$$\begin{aligned}\lambda \quad = \quad & \Big(C^3_{\frac{p-1}{2}} + C^3_{\frac{p+1}{2}} + C^2_{\frac{p+1}{2}} + \\ & (C^2_{\frac{p-1}{2}} + C^2_{\frac{p+1}{2}}) + C^1_{\frac{p+1}{2}}\Big) * \gamma^3\end{aligned} \tag{20}$$

And the probability of three disk failures is shown as Equation (21).

$$\lambda = C^3_{p+3} * \gamma^3 \tag{21}$$

Therefore, the proportion of non-recoverable cases of all three disk failures is shown as the following equation:

$$\Omega \quad = \quad \frac{\left(C^3_{\frac{p-1}{2}} + C^3_{\frac{p+1}{2}} + C^2_{\frac{p+1}{2}} + (C^2_{\frac{p-1}{2}} + C^2_{\frac{p+1}{2}}) + C^1_{\frac{p+1}{2}}\right)}{C^3_{p+3}}$$

From the above equation, it can be concluded that when the p tends to infinity, the probability of unable to

recover in the three disk failure is $\Omega = \frac{1}{4}$, that is to say, for the LREVENODD, 75% of the three disk failures can be recovered. And because in practice, the probability of three disk failures is much lower than the single disk failure, so the EVENODD extended by using the local repair algorithm is closer to MDS codes.

5 Performance Test

This section tests and compares the performance of the single disk failure recovery between EVENODD and LREVENODD. The machine parameters in the experimental environment cluster are: CPU Intel Core i7-3632, memory 8GB, disk 500GB, test file size 10M.

5.1 Encoding Test

Figure 11 gives the coding time of EVENODD and LREVENODD when the prime takes different values.

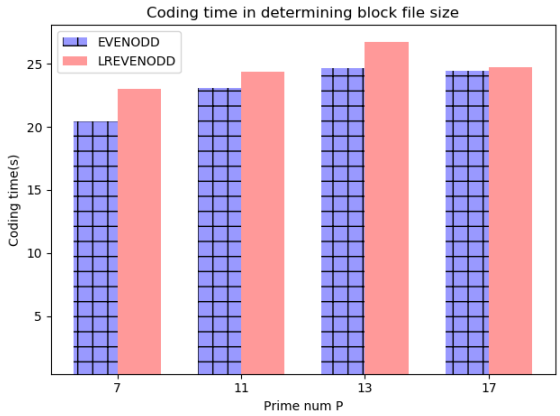


Figure 11: Coding time comparison

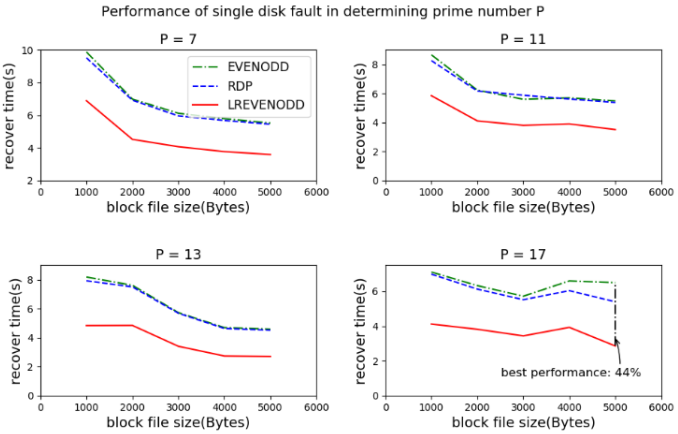


Figure 12: Single disk failure recover time in determing p

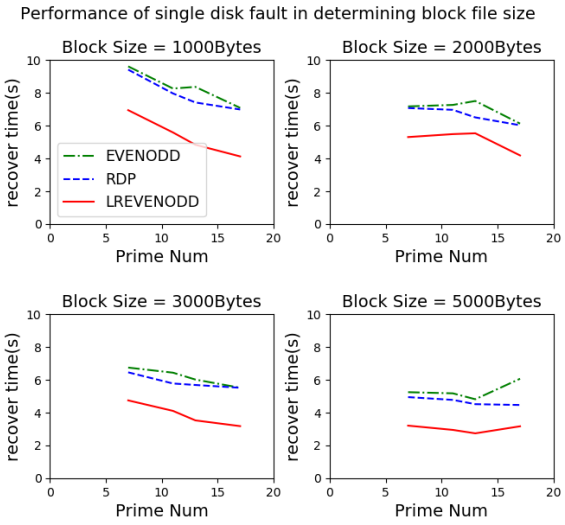


Figure 13: Single disk failure recover time in determing block file size

5.2 Single Disk Failure Recovery Test

Figure 12 gives a single disk failure recovery time when p is determined and block size is different. And Figure 13 gives a single disk failure recovery time when p takes different values.

Through the test results, it can be found that when the p increases and the block file size increases, the recovery overhead of single disk failure decreases. When p=17 and block file size is 5000Bytes, the overhead is reduced by 44%. And Figure 14 gives a comparison of the file block used in single disk failure recovery process between EVENODD and LREVENODD.

As shown in Figure 14, it can be found that when the number of file blocks increases, the number of data blocks required for single disk failure recovery in LREVENODD is closer to 50% of the number in EVENODD.

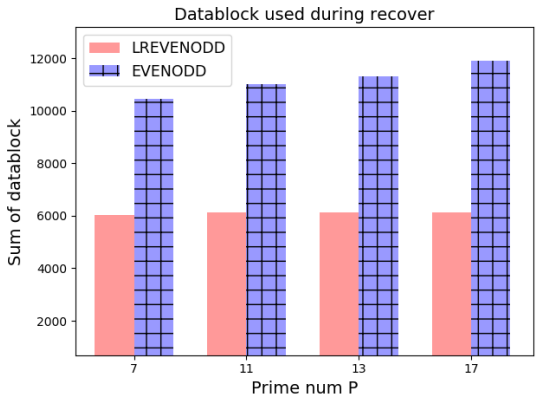


Figure 14: Single disk failure recovery reading overhead

5.3 Double Disk Failure Recovery Test

Figure 15 shows the recover time in double disk failure when the prime number p is different while block size is determined.

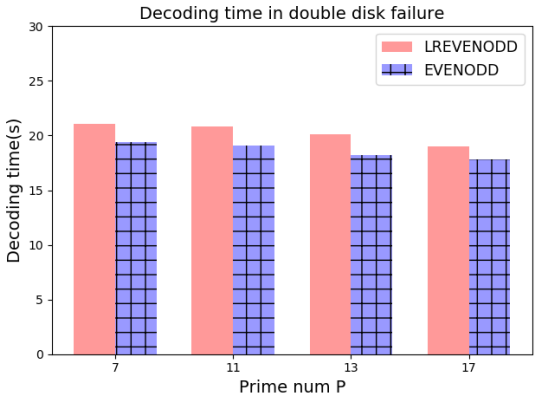


Figure 15: Double disk failures recover time

5.4 Triple Disk Failure Recovery Test

Figure 16 shows the unrecoverable times when 10000 triple disk failures occured with different prime number.

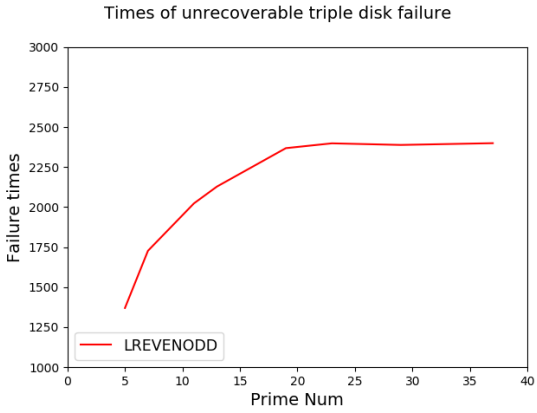


Figure 16: Times of unrecoverable triple disk failure

By Figure 16 knowable, when p increases, the number of non recoverable times is gradually approaching 2500 times, which is 25% of the total number of triple disk failure. Figure 17 shows the average recovery time of the triple disk failure when the file size is large and the prime number p takes different values.

6 Conclusions

Aiming at the problem that the read overhead of single disk failure recovery in EVENODD is too large, this paper proposes a method to reform it. Combining LRC with

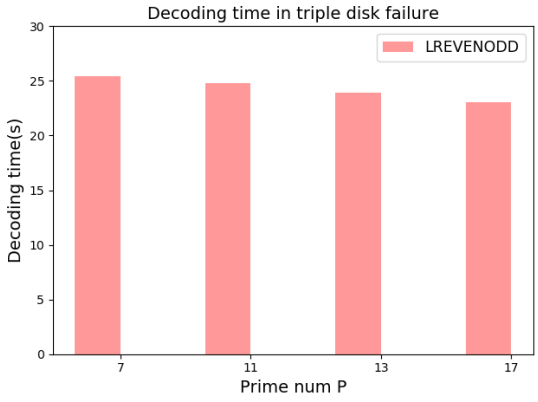


Figure 17: Decoding time in triple disk failure

RS codes, local repair method is used to improve EVENODD codes. The local repair method reduces the read overhead of single disk failure recovery by adding a new redundant column. The experimental results showed that the improved EVENODD code with local repair method can effectively reduce the read overhead of single disk failure recovery and improve the system performance. In fact, this method is not only applicable to EVENODD, but also can be used for RDP, STAR and RTP codes and many other slope codes, and the relevant proof and test need to be further completed.

Acknowledgments

This study was supported by the National Natural Science Fund of China 61501064 and Science and technology program of Sichuan 2018GZ0099. The authors gratefully acknowledge the anonymous reviewers for their valuable comments and teacher's hard cultivation.

References

[1] Q. Chang, Y. Xu, L. Xiang, and Y. Pan, "A hybrid recovery algorithm for single disk failure in evenodd (in chinese)," *Computer Applications and Software*, vol. 28, no. 6, pp. 15–18, 2011.

[2] L. Chen, D. Yuan, P. Teng, and X. Wang, "Inverse code: A low density mds horizontal array code that can accommodate 3 errors (in chinese)," *Journal of Sichuan University (Engineering Science Edition)*, no. 5, pp. 135–142, 2017.

[3] P. Corbett, B. English, A. Goel, T. Grcanac, S. Kleiman, J. Leong, and S. Sankar, "Row-diagonal parity for double disk failure correction," in *Usenix Conference on File and Storage Technologies*, pp. 1–1, 2004.

[4] A. Goel and P. Corbett, "Raid triple parity," *Acm Sigops Operating Systems Review*, vol. 46, no. 3, pp. 41–49, 2012.

[5] R. Hu, G. Liu, and J. Jiang, "An efficient coding scheme for tolerating double disk failures," in *IEEE International Conference on High Performance Computing and Communications*, pp. 707–712, 2010.

[6] Z. Huang, *Research on MDS Array Code in Fault-Tolerant Storage System*, PhD thesis, Huazhong University of Science and Technolog, 2016.

[7] C. Huang and L. Xu, "Star: An efficient coding scheme for correcting triple storage node failures," in *Conference on Usenix Conference on File and Storage Technologies*, pp. 15–15, 2005.

[8] M. Sathiamoorthy, M. Asteris, D. Papailiopoulos, A. G. Dimakis, R. Vadali, S. Chen, and D. Borthakur, "Xoring elephants: Novel erasure codes for big data," *Proceedings of the Vldb Endowment*, vol. 6, no. 5, pp. 325–336, 2013.

[9] D. Sun, *Research on Verification Update and Repair Optimization Technology for Fault-Tolerant Storage System*, PhD thesis, University of Science and Technology of China, 2017.

[10] P. Teng, J. Zhang, L. Chen, and X. Wang, "Random array code: A raid storage disaster recovery method with high disaster tolerance and scalability," *Journal of Sichuan University (Engineering Science Edition)*, vol. 49, no. 3, pp. 110–116, 2017.

[11] L. Xiang, Y. Xu, J. C. S. Lui, and Q. Chang, "Optimal recovery of single disk failure in rdp code storage systems," *Acm Sigmetrics Performance Evaluation Review*, vol. 38, no. 1, pp. 119–130, 2010.

[12] L. Xu and J. Bruck, "Highly available distributed storage systems," *A Caltech Library Service*, vol. 249, pp. 307–330, 1999.

Biography

Feng Xiao, master candidate. He is currently an Master student in Chengdu University of Information Technology, Chengdu, China. His research interests include coding theory and database theory.

Di Fan, master candidate. She is currently an Master student in Chengdu University of Information Technology, Chengdu, China. Her research interests include coding theory and information security.

Dan Tang received his Ph.D. degree from Graduate University of Chinese Academy of Sciences (CAS), Beijing, China in 2010. He is currently an associate professor with Chengdu University of Information Technology, Chengdu, China. His research interests include coding theory and secret sharing scheme.

A Pseudo Random Bit Generator Based on a Modified Chaotic Map

Chokri Nouar and Zine El Abidine Guennoun
(Corresponding author: Chokri Nouar)
Department of Mathematics, Mohamed V University in Rabat
No. 4, Avenue Ibn Battouta B. P. 1014 RP, Rabat, Morocco
(Email: corresponding chokri.nouar@gmail.com)
(Received Dec. 11, 2017; Revised and Accepted Apr. 8, 2018; First Online Jan. 14, 2019)

Abstract

This paper presents a new pseudo random bit generator based on a modified Gingerbreadman chaotic system, The new model has been well studied to avoid fixed and periodic points. We have verified that the system's Lyapunov exponent is positive, which means the system is a chaotic one. The randomness of the bits generated by the proposed generator is successfully tested by the NIST. We notice that during the execution of the algorithm, the password changes automatically after a number of iterations.

Keywords: Chaotic Systems; Gingerbreadman Map; Lyapunov Exponent; NIST; Pseudo-random Bit Generator

1 Introduction

In the mathematical theory, introduced by Devaney in 1988, the Gingerbreadman map is a simple two-dimensional chaotic map. Several studies, which were devoted to the Gingerbreadman map, show the existence of fixed and periodic points [4]. This property is undesirable by cryptography.

The idea of designing a pseudo-random bit generator by making use of chaotic first order nonlinear differential equation was proposed by Oishi and Inoue [9] in 1982. After their paper, several pseudo-random bit generators were suggested. Notice that the algorithms based on chaos showed a good performance for data encryption such as images, videos or audio data [8].

The aim of this paper is to propose a new pseudo-random bit generator based on a chaotic system. We will, firstly, try to eliminate the fixed and periodic points by adding a perturbation function. Then we will verify that the modified Gingerbreadman map is a chaotic system that includes appropriate features such as high sensitivity to initial conditions, unpredictability, mixing property and high complexity. This will allow the system to integrate into various cryptographic applications.

The rest of this paper is structured as follows: the first section presents how the chaotic system affects the produced sequences. In the second section, basic definitions of Lyapunov exponent and The Gingerbreadman map will be recalled. In the third section, we will test the modified Gingerbreadman chaotic system. In section four, we present a detailed description of our generator. Before concluding, the statistical analysis and validation of the bits sequences generated by our generator are given in section five.

2 Chaotic System

A chaotic system is a non-linear deterministic dynamical system, which exhibits pseudo-random behaviour. The output values of a chaotic system vary depending on specific parameters and initial conditions. Different parameter values yield different periods of oscillations at the output of the system.

Chaotic sequences produced by a chaotic system are pseudo-random ones, their structures are very complex and difficult to analyse and to predict. These sequences appear totally random to an external observer, in spite of their deterministic generation, as they are sensitively dependent on initial conditions. In other words, chaotic systems can improve the security of encryption systems.

In mathematics, "Lyapunov exponent" is a quantity that measures the speed at which those small differences are amplified; it actually measures the degree of sensitivity of chaotic systems. Those that have a chaotic behaviour are defined as a chaotic map [10].

The two following sections present a brief description of the Lyapunov exponent and the chaotic map (GingerBreadMan) used in this paper.

2.1 Lyapunov Exponent

In chaotic systems, the distance between two initially close trajectories tends to increase at an exponential speed and then stabilizes when the distance reaches a limit value.

The Lyapunov exponent is an approximate quantity that measures the exponential divergence of initially close trajectories, it also estimates the amount of chaos in any system.

Definition 1. *The Lyapunov exponent L computed using the derivative method is defined by*

$$L = 1/n(ln \mid f'(x_1) \mid + ln \mid f'(x_2) \mid + \cdots + ln \mid f'(x_n) \mid)$$

where f' represents differentiation with respect to x and $x_1, x_2 ... x_n$ are successive iterates. The Lyapunov exponent may be computed for a sample of points near the attractor to obtain an average Lyapunov exponent. [6]

Theorem 1. *If at least one of the average Lyapunov exponents is positive, then the system is chaotic; if the average Lyapunov exponent is negative, then the orbit is periodic. However when the average Lyapunov exponent is zero, a bifurcation occurs [7].*

2.2 The Gingerbreadman Map

In dynamical systems theory, the System (1) is called "the Gingerbreadman map." It was investigated by Devaney in 1984 as a simple module of a chaotic two-dimensional map presented by the following transformation:

$$\begin{cases} x_{n+1} &= 1 - y_n + \mid x_n \mid \\ y_{n+1} &= x_n \end{cases} \quad (1)$$

The Gingerbreadman map is a piecewise linear application, which has been shown to be chaotic in certain regions and stable in others. Figures 1 and 2 displays the first iterations of $(-0,2; 0,2)$.

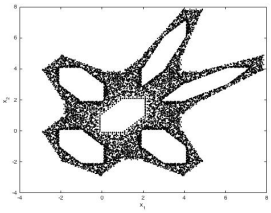


Figure 1: 10000 iterates

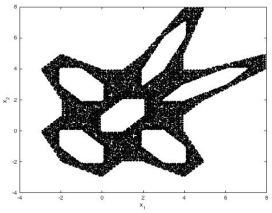


Figure 2: 20000 iterates

Despite its good property, algorithms [3] proved that the Gingerbreadman map has fixed and periodic points in the hexagonal as shown in Figure 3.

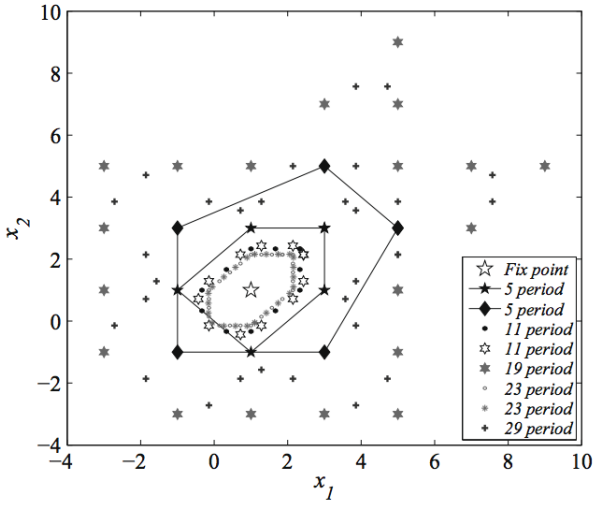


Figure 3: Periodic and fixed points of Gingerbreadman map

new model of Gingerbreadman map (NMGM) is given by the following system

$$H = \begin{cases} x_{n+1} &= 1 - y_n + \mid x_n \mid \\ y_{n+1} &= x_n + r \times \sin(y_n) \end{cases} \quad (2)$$

The aforementioned perturbation function is simple and periodic function to ensure that the dynamical system remains non-linear and deterministic; that is, it does not tend to infinity. It should be noted that it is possible to replace the $\sin(y_n)$ with $\cos(y_n)$ as shown in the Figures 4 and 5.

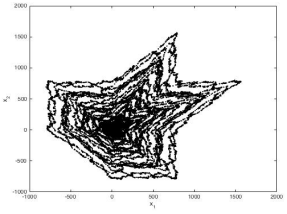


Figure 4: Gingerbreadman with $\cos(y)$ and $r = 3.8$

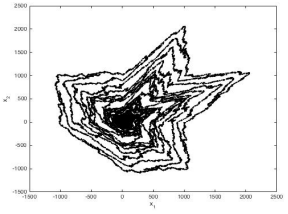


Figure 5: Gingerbreadman with $\sin(y)$ and $r = 3.8$

In the next section, we determine the intervals of the parameter r for which the system remains chaotic, by using the Lyapunov exponent.

3 A Proposed Chaotic System

3.1 Description of the Proposed System

In order to avoid these fixed and periodic points, we add the perturbation function $f(y_n) = r \times \sin(y_n)$ to the second equation in the Gingerbreadman system, where r is a non-arbitrarily chosen real parameter. Therefore, the

3.2 Chaotic Tests of the Proposed System

The Lyapunov exponent of the proposed system varies depending on the parameter r.

The Figure 6 gives the curve of the Lyapunov exponent in function of the parameter r. One can remark that the Lyapunov exponent of the new model of Gingerbreadman

map (NMGM) is positive when $r > 2$ for $x_0 = 0$ and $y_0 = 0$, which implies that the proposed system is chaotic.

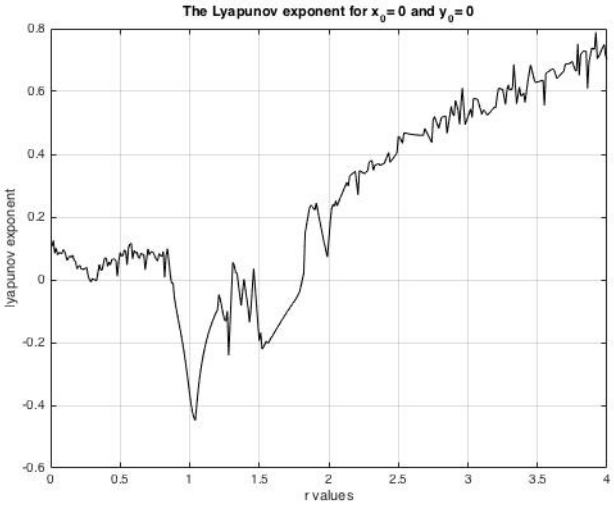


Figure 6: The Lyapunov exponent of the proposed system

Another simple method used to determine whether or not a system is chaotic, is to use the sensitivity to initial conditions. Figures 7, 8, 9 and 10 show how the attractors of the NMGM are affected by small differences in the initial conditions.

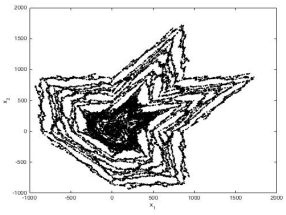


Figure 7: NMGM with $r = 3.8$, $y_0 = 0$ and $x_0 = 0.1$

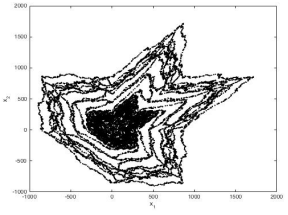


Figure 8: NMGM with $r = 3.8$, $y_0 = 0$ and $x_0 = 0.1 + 0.1 \times 10^{-6}$

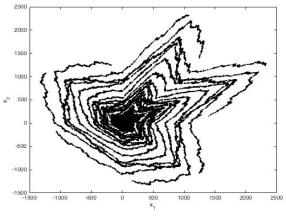


Figure 9: NMGM with $r = 3.8$, $y_0 = 0$ and $x_0 = 0.1 + 0.1 \times 10^{-9}$

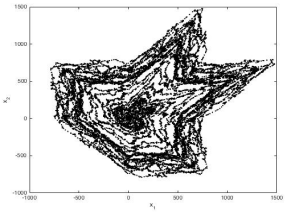


Figure 10: NMGM with $r = 3.8$, $y_0 = 0$ and $x_0 = 0.1 + 0.1 \times 10^{-12}$

3.3 Comparison of the Bifurcation and Lyapunov Exponent

In this section, the comparison between our system and Gingerbreadman map is presented. Figures 1 and 2 shows the bifurcation diagrams of Gingerbreadman map. It is apparent from comparison of Figures 7, 8, 9 and 10, the bifurcation of our system is well distributed.

Figures 11, 12, 13 and 14 shows the Lyapunov exponent of NMGM is more than 0.6 where the Lyapunov exponent of Gingerbreadman map is less than 0.15. So we can concluded that, the NMGM is more chaotic.

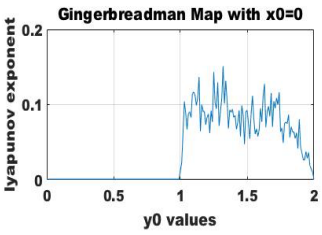


Figure 11: Gingerbreadman map y_0

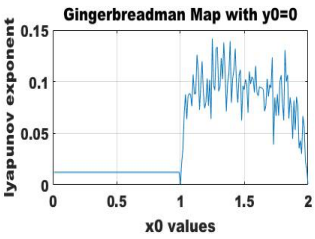


Figure 12: Gingerbreadman map x_0

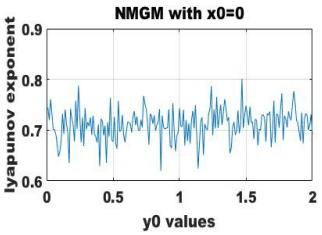


Figure 13: NMGM with y_0

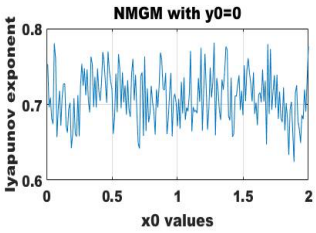


Figure 14: NMGM with x_0

4 Designing a PRBG Based on the NMGM

Our pseudo-random bit generator based on the new model of Gingerbreadman map (PRBG-NMGM) is a deterministic generator initialized by a password Pw of any size, whose output is a cryptographically secure binary sequence.

The initial conditions of our system (NMGM) x_0, y_0 and r_0 are calculated from the password Pw by a method based on a pointer that is positioned on a bit of the Pw. The pointer moves from a position to another according to a linear congruential throughout the ASCII representation of the Pw [2].

After a predetermined number of iterations performed for the three initial conditions x_0, y_0 and r_0, we start generating the numbers needed to construct the final sequence $S = S_1 ... S_n$ with $S_i = X_i \oplus Y_i$, where X_i and Y_i are two 32-bit numbers generated in the i^{th} step.

4.1 The Calculation of the Initialization Values x_0 and y_0

From a binary string of any length n which represents the password $Pw = (P_1P_2...P_n)_2$, we calculate the three initial conditions x_0,y_0 and r_0. For that we extracted 64 bits for each value from the Pw.

We consider a pointer (Z_i) that takes values indicating the bit positions of Pw. The sequences of positions are defined as follows

$$\begin{cases} Z(0) &= \lfloor n/4 \rfloor \\ Z(i+1) &= ((n^2+1) \times Z(i)+1) mod(n) \end{cases} \quad (3)$$

The pointer moves throughout the password and returns to 64 bits stream length, which are classified as follows PT=$(P_{Z(0)}P_{Z(1)}P_{Z(2)}P_{Z(3)}...P_{Z(61)}P_{Z(62)}P_{Z(63)})$.

We calculate the number A B and C from PT ($0 \leq A, B, C < 2^{64}$) as follows:

$$\begin{aligned} A &= (\overline{P_{Z(0)}}P_{Z(1)}P_{Z(2)}\overline{P_{Z(3)}}...P_{Z_{(61)}}P_{Z_{(62)}}\overline{P_{Z(63)}})_2 \\ &= \sum_{i=0}^{21} \overline{P_{Z(3i)}} \times 2^{63-3i} + \sum_{i=0}^{20} P_{Z(3i+1)} \times 2^{63-3i-1} \\ &\quad + \sum_{i=0}^{20} P_{Z(3i+2)} \times 2^{63-3i-2} \quad (4) \\ B &= (P_{Z(0)}\overline{P_{Z(1)}}P_{Z(2)}P_{Z(3)}...\overline{P_{Z_{(61)}}}P_{Z_{(62)}}P_{Z(63)})_2 \\ &= \sum_{i=0}^{20} P_{Z(3i)} \times 2^{63-3i} + \sum_{i=0}^{21} \overline{P_{Z(3i+1)}} \times 2^{63-3i-1} \\ &\quad + \sum_{i=0}^{20} P_{Z(3i+2)} \times 2^{63-3i-2} \quad (5) \\ C &= (P_{Z(0)}P_{Z(1)}\overline{P_{Z(2)}}P_{Z(3)}...P_{Z(61)}\overline{P_{Z(62)}}P_{Z(63)})_2 \\ &= \sum_{i=0}^{20} P_{Z(3i)} \times 2^{63-3i} + \sum_{i=0}^{20} P_{Z(3i+1)} \times 2^{63-3i-1} \\ &\quad + \sum_{i=0}^{21} \overline{P_{Z(3i+2)}} \times 2^{63-3i-2} \quad (6) \end{aligned}$$

Finally, $x_0 = \frac{A}{2^{63}}, y_0 = \frac{B}{2^{63}}$ and $r_0 = \frac{C}{2^{63}} + 2$. So the initial conditions values are in the fallowing intervals: $0 \leq x_0 < 2$, $0 \leq y_0 < 2$ and $2 \leq r_0 < 4$.

Algorithm 1 is used to calculate initial values x_0, y_0 and r_0

4.2 Generating the Pseudo-random Sequence

After extracting the initial values x_0, y_0 and r_0, the system (PRBG-NMGM) will be ready to generate the pseudo random bits sequences.

Algorithm 2 of generating has two input parameters, a password Pw and an integer F that indicates the length of the output binary sequence.

Algorithm 1 Initialization

Input password $Pw = (P_1P_2...P_n)_2$ a binary string of any length n
Output initiation values x_0, y_0 and r_0
$Z \leftarrow \lfloor n/4 \rfloor$
$A, B, C \leftarrow 0$
for $i \leftarrow 0$ **to** 63 **do**
  $Z \leftarrow ((n^2+1) \times Z + 1) mod(n)$
  if i mod (3) $= 0$ **then**
    $A \leftarrow A + \overline{P_Z} \times 2^{63-i}$
    $B \leftarrow B + P_Z \times 2^{63-i}$
    $C \leftarrow C + P_Z \times 2^{63-i}$
  else
    if i mod (3) $= 1$ **then**
      $A \leftarrow A + P_Z \times 2^{63-i}$
      $B \leftarrow B + \overline{P_Z} \times 2^{63-i}$
      $C \leftarrow C + P_Z \times 2^{63-i}$
    else
      $A \leftarrow A + P_Z \times 2^{63-i}$
      $B \leftarrow B + P_Z \times 2^{63-i}$
      $C \leftarrow C + \overline{P_Z} \times 2^{63-i}$
    end if
  end if
end for
$x_0 \leftarrow \frac{A}{2^{63}}$; $y_0 \leftarrow \frac{B}{2^{63}}$; $r_0 \leftarrow \frac{C}{2^{63}} + 2$;
return $x_0; y_0; r_0$

Step 1: In the first step leaving the system NMGM looping up to n_0 iterations to avoid the harmful effects of transitional procedures [1], where n_0 is determined from the length of Pw such as

$$n_0 = r_0 \times length(Pw).$$

Step 2: The iteration of the system NMGM continues, for each $i \leq F$ and $mod(i, length(Pw)) \neq 0$ we obtain the pairs (x_i, y_i) to construct the sub-sequence $S_i = X_i \oplus Y_i$ such as

$$X_i = floor(mod(x_i, 1) \times 2^{32}),$$
$$Y_i = floor(mod(y_i, 1) \times 2^{32}),$$

where the $floor(x)$ returns the largest integer less than or equal to x, $mod(x, y)$ returns the reminder after division of x by y, X_i and Y_i are two 32-bit numbers generated in the i^{th} step, and $\oplus$ represents operator of exclusive-OR.

Step 3: If $mod(i, length(Pw) = 0$ the parameter r_j of the system NMGM change automatically, we obtain the new r_j with the following equation:

$$r_{j+1} = (r_j + 1)^2 mod(2) + 2,$$

else return to step 2 until the bit stream limit is reached.

The output S of the PRBG-NMGM is the concatenation of the sub-sequences $S_1S_2...S_i...S_F$. Figure 15 shows scheme of i^{th} generation step of our PRNG-NMGM.

Algorithm 2 Generation

Input password $Pw = (P_1P_2...P_n)_2$ and F the length of the requested binary sequence
Output S the random binary sequence
$x, y, r \leftarrow$ Initialize(Pw)
$M \leftarrow r \times length(Pw)$
$S \leftarrow 0$
$i, j, k \leftarrow 0$
while $j \leq F$ **do**
  if $i \leq M$ **then**
    $(x_{i+1}, y_{i+1}, r) \leftarrow H(x_i, y_i, r)$
    $i \leftarrow i + 1$
  else
    if $i\%n \neq 0$ **then**
      $(x_{i+1}, y_{i+1}, r) \leftarrow H(x_i, y_i, r)$
      $X \leftarrow \lfloor (x_{i+1} mod(1) \times 2^{32} \rfloor$
      $Y \leftarrow \lfloor (y_{i+1} mod(1) \times 2^{32} \rfloor$
      $R = X \bigoplus Y$
      $S \leftarrow S \parallel R$
      $i \leftarrow i + 1$
      $j \leftarrow j + 1$
    else
      $r \leftarrow (r+1)^2 mod(2) + 2$
      $i \leftarrow i + 1$
    end if
  end if
end while
return S

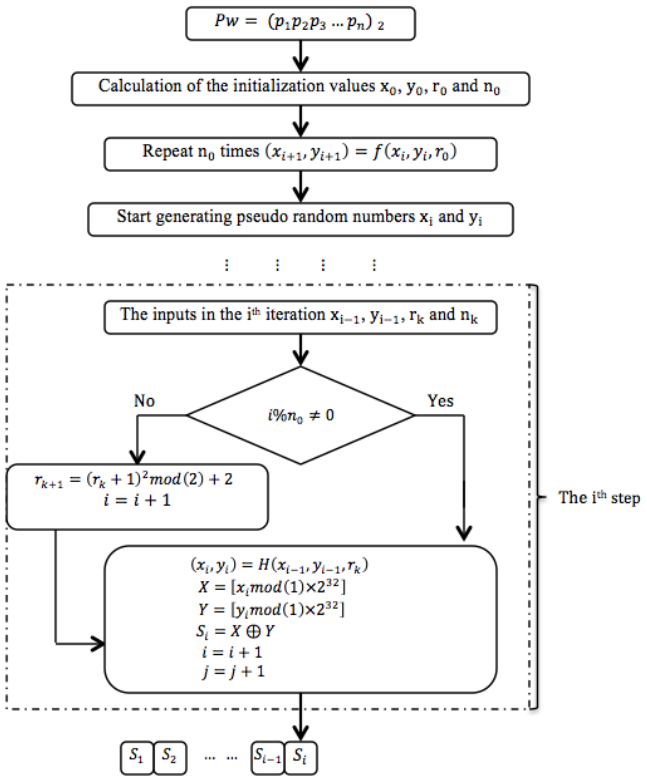


Figure 15: Scheme of i^{th} generation step of our PRNG

5 Security Analysis

Security analysis is a tool for evaluating the performance of our proposal PRBG-NMGM. In this section we examine the key space size, the sensitivity to the initial conditions and the randomness level of the generated sequences.

5.1 Key Space

Among the most important criteria of a cryptosystem is the size of key space. A space large enough security keys make exhaustive attacks infeasible. Our PRBG-NMGM initialized by a key of any size as already mentioned.

On the other hand a key space of size larger than 2^{128} is computationally secured against exhaustive attacks [5], therefore, the key size N must be greater than 128.

The calculation of the three initial values x_0, y_0 and r_0 needs exactly 192 bits, which are extracted via a pointer that traverses the binary string, so the space of initial values is 2^{192}. This leads us to say that the size of the key space is large enough to be attacked exhaustively.

5.2 Key Sensitivity

The key sensitivity implies that the small change in the secret key should produce a big change in the pseudo-random sequences, this property is essential to make a highly secured PRBG against statistical and differential attacks. This property is also basic for the PRGB not to be broken even if there is a small difference between the keys. The proposed generator is based on a chaotic map of the positive Lyapunov exponent meaning that is very sensitive to the initial conditions.

In order to examine the security of our generator, we have performed the key sensitivity test; we place several keys k_i in the input of the generator with a bit of difference between them, then we calculate the hamming distance between two pseudo-random sequences S_i of the size N generated by each key.

The calculation of the hamming distance between two binary sequences is the number $DH(S_i, S_j) = card\{e/x_e \neq y_e\}$ with $S_i = x_1x_2...x_N$ and $S_j = y_1y_2...y_N$. In general, this distance is given by:

$$DH(S_i, S_j) = \sum_{k=1}^{N} (x_k \bigoplus y_k).$$

The fact that a generator is very sensitive to the key makes the hamming distance vary in the neighbourhood of $N/2$, resulting the $DH(S_i, S_j)/N$ being about 0.50 for each pair of sequences produced.

In the next test we will generate a set of pseudo-random sequences $\{S_i\}$ from the keys $\{k_i\}_{0 \leq i \leq 55}$.

We consider $k_0 =$ "$ABIDINE$" whose binary representation in ASCII code is $k_0 = (01000001\ \ 01000010\ \ 01001001\ \ 01000100\ \ 01001001\ \ 01001110\ \ 01000101)_2$. The other 55 keys $\{k_i\}_{1 \leq i \leq 55}$ are derived from the k_0,

modifying the i^{th} bit among the 56 bits of the k_0 to find k_i.

The result of the Hamming distance between the sequences is given in Figure 16.

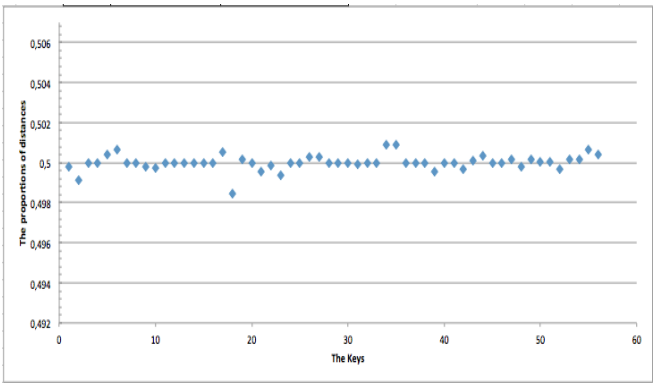


Figure 16: The Hamming distance between the sequences

It is clear that the proportions of difference between the sequences are about 50%, which implies that the proposed generator is purely sensitive to the initial conditions.

This sensitivity is due to the chaotic system that constructs the generator, meaning that the generated sequences are chaotic and unpredictable. It is also due to the initialization values of the PRBG-NMGM that are derived from a password, using a method based on a Linear Congruential Generators.

Indeed, a change of one bit between two keys leads to a totally different initialization values as well as different generated sequences.

5.3 Randomness Level

We used the NIST tests and DIEHARD tests in order to measure the level of randomness of the bits sequences generated by PRBG-NMGM .

The NIST tests suite consists of 15 tests developed to quantify and to evaluate the degree of randomness of the binary sequences produced by the cryptographic generators.

These tests are: frequency (monobit), block-frequency, cumulative sums, runs, longest run of ones, rank, Fast Fourier Transform (spectral), non-overlapping templates, overlapping templates, Maurers Universal Statistical, approximate entropy, random excursions, random-excursion variant, serial, and linear complexity.

For each statistical test, a P_{value} is calculated from the bit sequence. This P_{value} is compared to a predefined threshold $\alpha = 0.01$, which is also called significance level.

If P_{value} is greater than 0.01, then the sequence is considered to be random, and it proceeds the statistical test successfully. Otherwise, the sequence does not appear random.

To apply the NIST tests on our generator we generated 1000 sequences, the size of each sequence is 10^6 bit from a different key. The table 1 bellow presents the test results in the sequences.

Table 1: NIST statistical test suite results for 1000 sequences of size 10^6 bit each generated by the our generator

NIST statistical test	P_{value}	Pass rate
Frequency	0,800005	989/1000
Block-Frequency	0,100709	990/1000
Cumulative Sums	0,233162	989/1000
Runs	0,350485	994/1000
Longest Run	0,719747	989/1000
Rank	0,402962	995/1000
FFT	0.345650	987/1000
Non-Overlapping	0,509841	990/1000
Overlapping	0.709558	987/1000
Universal	0.390721	990/1000
Approximate Entropy	0.846338	986/1000
Random Excursions	0,338148	620/628
Random Excursions Variant	0,592461	623/628
Serial	0,490572	990/1000
Linear Complexity	0.805569	987/1000

The minimum pass rate for each statistical test with the exception of the random excursion (variant) test is approximately 980 for a sample size 1000 binary sequences.

The minimum pass rate for the random excursion (variant) test is approximately 613 for a sample size 627 binary sequences.

The DIEHARD tests consists of a set of statistical tests for measuring the quality of randomness, developed by George Marsaglia, these tests are: birthday spacings, overlapping 5-permutations, binary rank (31 x 31), binary rank (32 x 32), binary rank (6 x 8), bitstream, Overlapping-Pairs-Sparse-Occupancy, Overlapping-Quadruples-Sparse-Occupancy, DNA, stream count- the-ones, byte-count-the-ones, 3D spheres, squeeze, overlapping sums, runs up, runs down, craps.

For the DIEHARD tests, we generated 1000 sequences of one million bits each, by the proposed pseudo-random bit generators. The results are given in Table 2.

We can see from **Table 1** that the NIST tests suite is passed successfully. The p_{value} of all tests is greater than the minimum rate (0.01).

Table 2 shows the DIEHARD P_{values} are in acceptable range of $(0, 1)$, and all tests are passed successfully.

Based on these results, the NMGM based random generator is suitable for cryptographic applications.

6 Conclusion

In this paper, a novel pseudo-random bits generator based on modified chaotic map was presented.

Table 2: DIEHARD statistical test suite results for 1000 sequences of size 10^6 bit each generated by the our generator

DIEHARD test name	P_{value}	Assessment
Birthday	0.92580677	passed
Overlapping 5-permutation	0.22867882	passed
Binary rank (32 x 32)	0.82601210	passed
Binary rank (6 x 8)	0.59379048	passed
Bitstream	0.90091209	passed
OPSO	0.09739253	passed
OQSO	0.54519450	passed
DNA	0.17853645	passed
Stream count-the-ones	0.50254509	passed
Byte count-the-ones	0.39753149	passed
Parking lot	0.91507220	passed
Minimum distance	0.91222820	passed
3D spheres	0.41362890	passed
Squeeze	0.63363326	passed
Runs up	0.09807447	passed
Runs down	0.49918763	passed
Craps	0.61161595	passed

The new model has been selected after a rigorous analysis that showed high dimensional chaotic which generate more complex and unpredictable chaotic sequences.

The results of statistical analyses like randomness, key space and key sensitivity indicate high security and suitability of the proposed generator for practical encryption.

In future works, we will apply our proposal PRBG-NMGM especially in audio and image encryption and in other cryptographic applications

References

[1] S. Borislav and K. Krasimir, "Novel secure pseudo-random number generation scheme based on two tinkerbell maps," *Advanced Studies in Theoretical Physics*, vol. 9, no. 9, pp. 411–421, 2015.

[2] K. Charif, A. Drissi, and Z. E. A. Guennoun, "A pseudo random number generator based on chaotic billiards," *International Journal of Network Security*, vol. 19, pp. 479–486, May 2017.

[3] G. Feiab, F. Feng-xiaa, D. Yan-fanga, Q. Yi-boa, and I. Balasingham, "A novel non-lyapunov approach through artificial bee colony algorithm for detecting unstable periodic orbits with high orders," *Expert Systems with Applications*, vol. 39, pp. 12389–12397, Nov. 2012.

[4] F. Gao, H. Gao, Z. Li, H. Tong, and J. J. Lee, "Detecting unstable periodic orbits of nonlinear mappings by a novel quantum-behaved particle swarm optimization non-lyapunov way," *Chaos, Solitons and Fractals*, vol. 42, pp. 2450–2463, Nov. 2009.

[5] E. Hato and D. Shihab, "Lorenz and rossler chaotic system for speech signal encryption," *International Journal of Computer Applications*, vol. 128, pp. 25–33, Oct. 2015.

[6] M. B. Jacques, C. Jean-François, and G. Christophe, "Quality analysis of a chaotic proven keyed hash function," *International Journal On Advances in Internet Technology*, Aug. 2016. (`https://arxiv.org/abs/1608.05928`)

[7] S. Lynch, *Dynamical Systems with Applications Using MATLAB*, Boston, USA: Birkhuser, 2014. (`https://www.springer.com/us/book/9783319068190`)

[8] A. Musheer, A. Bashir, and F. Omar, "Chaos based mixed keystream generation for voice data encryption," *International Journal on Cryptography and Information Security*, vol. 2, no. 1, pp. 36–45, 2014.

[9] S. Oishi and H. Inoue, "Pseudo-random number generators and chaos," *IEICE Transactions*, vol. E65, pp. 534–541, September 1982.

[10] R. Swati and T. Sanjeev, "Security analysis of multimedia data encryption technique using piecewise linear chaotic maps," *International Journal on Recent and Innovation Trends in Computing and Communication*, vol. 1, pp. 458–461, May 2013.

Biography

Chokri NOUAR He is received his Master's degree in mathematics and statistics, option cryptography and information security from Mohammed-V University in Rabat, Morocco. He is actually a PhD student in the Laboratory of Mathematics, statistic and applications. His major research interests include information security and cryptography.

Zine El Abidine GUENNOUN He is a full professor of Department of Mathematics at the Faculty of Science, Mohamed V University in Rabat, Morocco. He received his Ph.D. (1989). His research interests include non linear analysis, fixed point theory, differential equation, financial mathematics and cryptography.

Group-Wise Classification Approach to Improve Android Malicious Apps Detection Accuracy

Ashu Sharma and Sanjay Kumar Sahay
(Corresponding author: Sanjay K. Sahay)

Birla Institute of Technology and Science, Pilani, Department of Computer Science and Information Systems
Goa Campus, NH-17B, By Pass Road, Zuarinagar-403726, Goa, India
(Email: ssahay@goa.bits-pilani.ac.in)

(Received Dec. 21, 2017; Revised and Accepted Mar. 7, 2018; First Online Jan. 14, 2019)

Abstract

In the fast-growing smart devices, Android is the most popular OS, and due to its attractive features, mobility, ease of use, these devices hold sensitive information such as personal data, browsing history, shopping history, financial details, *etc.* Therefore, any security gap in these devices means that the information stored or accessing the smart devices are at high risk of being breached by the malware. These malware are continuously growing and are also used for military espionage, disrupting the industry, power grids, *etc.* To detect these malware, traditional signature matching techniques are widely used. However, such strategies are not capable to detect the advanced Android malicious apps because malware developer uses several obfuscation techniques. Hence, researchers are continuously addressing the security issues in the Android based smart devices. Therefore, in this paper using *Drebin* benchmark malware dataset we experimentally demonstrate how to improve the detection accuracy by analyzing the apps after grouping the collected data based on the permissions and achieved 97.15% overall average accuracy. Our results outperform the accuracy obtained without grouping data (79.27%, 2017), Arp, *et al.* (94%, 2014), Annamalai *et al.* (84.29%, 2016), Bahman Rashidi *et al.* (82%, 2017)) and Ali Feizollah, *et al.* (95.5%, 2017). The analysis also shows that among the groups, *Microphone* group detection accuracy is least while *Calendar* group apps are detected with the highest accuracy, and for the best performance, one shall take 80-100 features.

Keywords: Android Malicious Apps; Dangerous Permissions; Machine Learning; Static Malware Analysis

1 Introduction

The attractive features and mobility of smart devices have drastically changed the today's environment. Many functionalities of these devices are similar to the traditional information technology system, which can also access enterprises applications and data, enabling employees to do their work remotely. Hence the security risks are not only limited to Bring Your Own Smart Device (BYOSD) scenarios but also for the devices which are adopted on an ad hoc basis. Therefore, any security gap in these devices means that the information stored or accessing smart devices are at high risk of being breached. The recent attack shows that the security features in these devices are not as par to completely stop the adversary [23]. Hence smart devices are becoming an attractive target for the online criminal, and they are investing more and more for the sophisticated attacks viz. ransomware or to steal the valuable personal data from the user device.

In the smart devices, Android is the most popular operating systems and are connected through the internet accessing billions of online websites (an estimate shows that 5 out of 6 mobile phones are working on Android OS [25]). Its popularity is basically due to its open source, exponential increase in the Android supported apps, third-party distribution, free rich SDK and the very much suited Java language. In this growing Android apps market, it is very hard to know which apps are malicious. As per Statista [24], there are approximately two million apps at the *Play Store* of Google and also many third-party apps available for the Android users. Hence potential of the malicious apps or malware entering these systems is now at never seen before levels, not only to the normal users but also for military espionage, disrupting the industry, power grids (e.g., Duqu, StuxNet), *etc.* [21]. In this, Quick Heal Threat Research Labs in the 3rd quarter of 2015 reported that they had received $\sim 4.2 \times 10^5$ malware per day for the Android and Windows platforms [15].

To detect the malware, traditional approaches are based on the signature matching, which is efficient from a time perspective but not relevant for the detection of advanced malicious apps and continuously growing zero-day malware attack [9]. Also, to evade the signature-based techniques, malware developer uses several obfuscation techniques. However, to detect the Android malicious apps, time to time, a number of static and dynamic meth-

ods have been proposed [2,5,11,16]. But, it appears that the proposed methods are not good enough to effectively detect the advanced malware [21] in the fast-growing internet and Android based smart devices usage into our daily life. Hence researchers are continuously addressing the security issues in the Android based smart devices. Therefore, in this paper, for the effective detection of Android malicious apps with high accuracy, we classified the apps after grouping the collected data based on permissions. The remaining paper is organized as follows. In next Section, we discuss the related work. Section 3 describes how the collected Android apps are grouped, Section 4 explains the feature selection approach, while Section 5 describes our approach for the effective detection of Android malicious apps and the obtained experimental results. Finally, Section 6 contains the conclusion.

2 Related Work

In both the two main methods (static and dynamic) used for the classification of malicious apps, selected classifiers are trained with a known dataset to differentiate the benign and malicious apps. In this, Arpil *et al.* achieved 94% detection accuracy by generating a joint vector space using AndroidManifest.xml file and the disassembled code [2]. Seo, *et al.* also used the same static features viz. permissions, dangerous APIs, and keywords associated with malicious behaviors to detect potential malicious apps [19].

Based on a set of characteristics derived from binary and metadata Gonzalez, *et al.* proposed a method *DroidKin*, which can detect the similarity among the apps under various levels of obfuscation [6]. Quentin *et al.*, uses op-code sequences to detect the malicious apps. However, their approaches are not suitable to detect the malware which are completely different [8].

In 2015, Smita Naval, *et al.* proposed an approach by quantifying the information-rich call sequences to detect the malicious binaries and claimed that the model is less vulnerable to call-injection attacks [12]. In 2016, Jaewook jang, *et al.* proposed *Andro-Dumpsys*, a hybrid malware detection approach based on the similarity between the malware creator-centric and malware-centric information. Their experimental analysis shows that *Andro-Dumpsys* can classify the malware families with good True Positive (TP) and True Negative (TN), and are also capable of identifying zero-day threats [7]. Luca Caviglione, *et al.* obtained 95.42% accuracy using neural networks and decision trees [12].

Sanjeev Das, *et al.* proposed *GuardOl* (a hardware-enhanced architecture), a combined approach using processor and field programmable gate array for online malware detection. Their approach detects 46% of malware for the first 30% of execution, while 97% on complete execution [4]. Saracino, *et al.*, proposed a host-based malware detection system called MADAM which simultaneously analyzes and correlates the features at four levels to detect the malware [18]. Gerardo Canfora, *et al.* analyzed two methods to detect Android malware, first was based on Hidden Markov Model, while the 2nd one exploits structural entropy and found that the structural entropy can identify the malware family more correctly [3].

Annamalai *et al.* proposed *DroidOl* for the effective online detection of malware using passive-aggressive classifier and achieved an accuracy of 81.29% [11].

Recently in 2017, Feizollah, *et al.* evaluated the effectiveness of Android Intents (explicit and implicit) as a distinguishing feature for identifying malicious applications. They conducted experiments using a dataset containing 7406 applications comprising 1846 clean and 5560 infected applications. They achieved the detection rate of 91% using Android Intent and 83% using Android permission. With the combination of both the features, they have achieved 95.5% detection rate [5]. Nikola *et al.* estimated F-measure (*does not take account of correctly classified benign apps*) of 95.1% and 89% by classifying the apps based on source code and permission respectively [10].

Rashidi *et al.* experimented with the *Drebin* benchmark malware dataset and shown that their model can accurately assess the risk levels of malicious applications and provide adaptive risk assessment based on user input and can find malware with the maximum accuracy of 82% [16].

3 Grouping of Android Apps

In Android, apps run as a separate process with unique user/group ID and operate in an application sandbox so that apps execution can be kept in isolation from other apps and the system. Hence, to access the user data/resources from the system, apps need additional capabilities that are not provided by the basic sandbox. To access data/resources which are outside of the sandbox, the apps have to explicitly request the needed permission. Depending on the sensitivity of data/area, requested permission may be granted automatically by the system or ask the user to approve or reject the request. In Android, these permissions can be found in Manifest.permission file e.g. to use the call service in an Android app, it should specify:

$< manifest xmlns :$
$Android =$ "$http : //schemas.Android.com/apk/res/Android$"
$package =$ "$com.Android.app.callApp$" $>$
$< uses - permissionAndroid :$
$name =$ "$android.permission.CALL\_PHONE$"$/ >$
...
$< /manifest >$

In total there are 235 permissions out of which 163 are hardware accessible and remaining are for user information access [13]. In terms of security, all these permissions can be put into two categories i.e. normal and dangerous permissions [1]. Therefore it will be important to study

the classification of Android malicious apps after grouping them into dangerous and normal/other permissions (Table 1). Hence in this paper to improve the overall average detection accuracy of Android malicious apps we use *Drebin* [2] 5531 benchmark malware dataset and 4235 benign apps available at Google play store. Our analysis shows that the *Drebin* dataset does not contain any apps which need body sensors permission.

Therefore we ignored the Sensors group in our experimental analysis and made total nine groups (eight groups of dangerous permissions and one group of normal/other permissions) for the detection of Android apps.

Table 1: Dangerous permissions groups of the Android apps

Group	Permissions
Calendar	Read calendar and write calendar.
Camera	Use camera.
Contacts	Read contacts, write contacts and get contacts.
Location	Access fine location and Access coarse location.
Microphone	Record audio.
Phone	Read phone state, call phone, read call logs, add voicemail, use sip and process outgoing calls.
Sensors	Use body sensors
SMS	Send SMS, receive SMS, read SMS receive WAP push and receive MMS.
Storage	Read external storage and write external storage.

4 Feature Selection

For the detection of Android malicious apps, feature selection plays a vital role, not only to represent the target concept but also to speed-up the learning and testing process. In this, often datasets are represented by many features. However, few of them may suffice to improve the concept quality, and also limiting the features will speed-up the classification. The Android apps can be represented as a vector of 256 opcodes [14], and some of these opcodes can be used as features for the effective and efficient detection of Android malicious apps. Therefore, to find the prominent features which can represent the target concept, opcodes from the collected Android apps are extracted as follows

- The *.apk* files (Android apps) has been decompiled by using freely available *apktool*;
- From the decompiled data, we kept only the *.smali* files and discarded other data, and then;
- Opcodes are extracted from the *.smali* files.

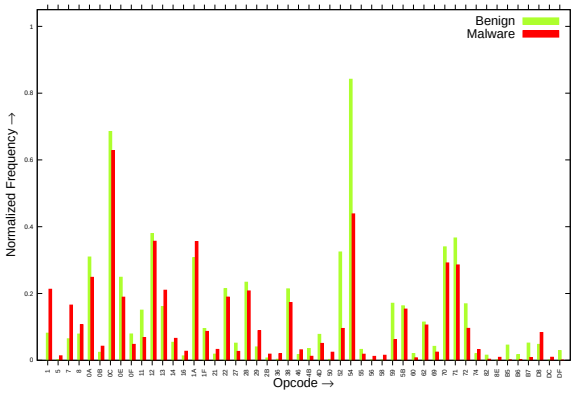


Figure 1: Top 50 opcodes occurrence difference between benign and malicious apps in the Calendar group

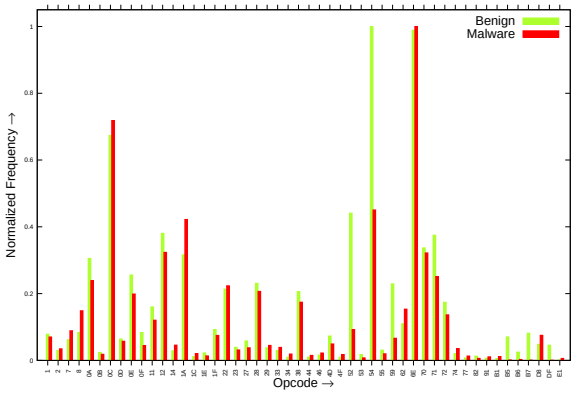


Figure 2: Top 50 opcodes occurrence difference between benign and malicious apps in the Camera group

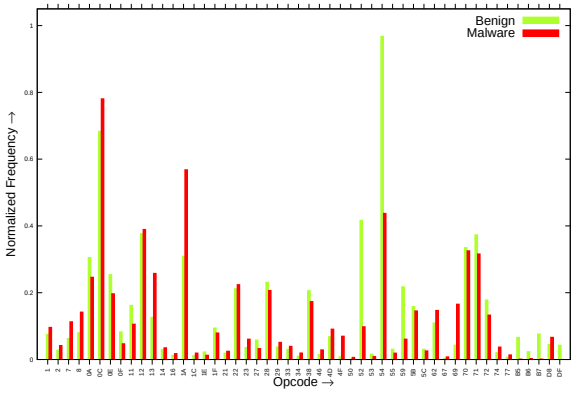


Figure 3: Top 50 opcodes occurrence difference between benign and malicious apps in the Contacts group

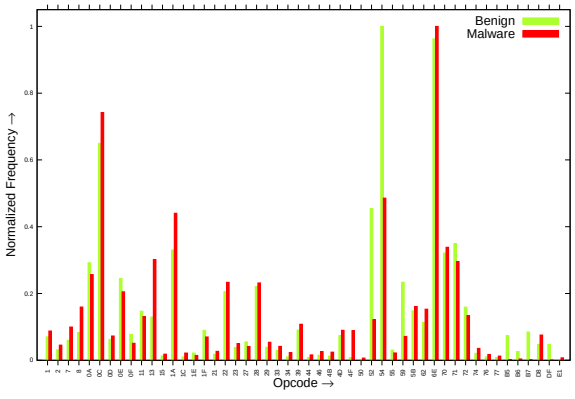


Figure 4: Top 50 opcodes occurrence difference between benign and malicious apps in the Location group

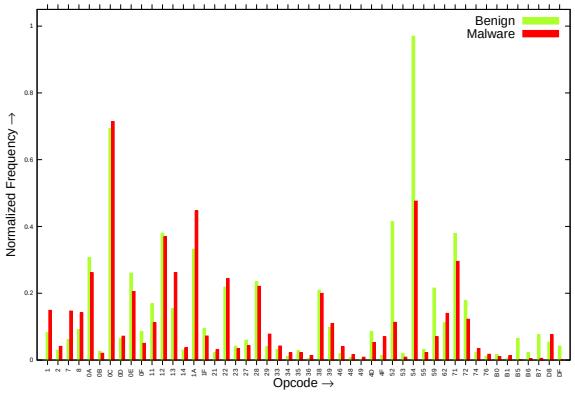


Figure 5: Top 50 opcodes occurrence difference between benign and malicious apps in the Microphone group

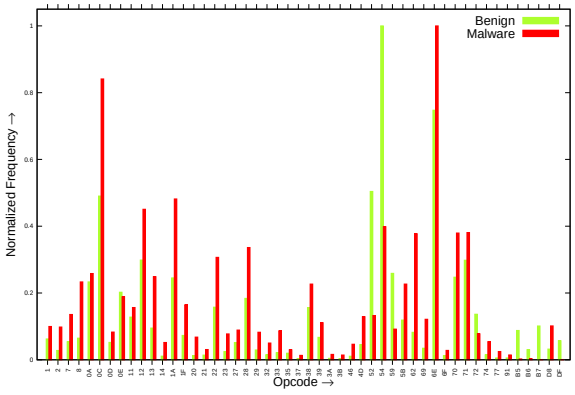


Figure 6: Top 50 opcodes occurrence difference between benign and malicious apps in the Other group

We studied the occurrence of opcodes in benign and malicious apps separately in each formed group, and computed the opcode occurrences difference between them. We observe that the opcode occurrence between malicious and benign apps among the formed group differ significantly (group-wise top 50 opcodes whose occurrence significantly differ are shown in Figures 1 - 9 for the *Calendar, Camera, Contacts, Location, Microphone, Others, Phone, SMS,* and *Storage* group respectively). Also, we find that the opcode occurrence in any group differs significantly when compared with the opcode occurrence obtained without forming the groups [22].

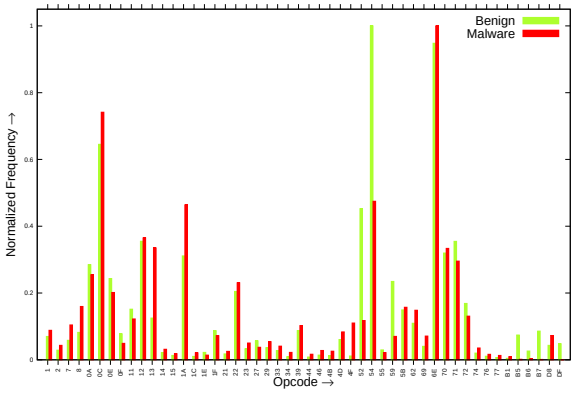


Figure 7: Top 50 opcodes occurrence difference between benign and malicious apps in the Phone group

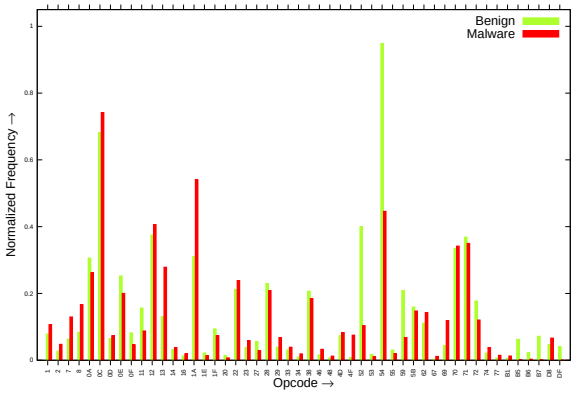


Figure 8: Top 50 opcodes occurrence difference between benign and malicious apps in the SMS group

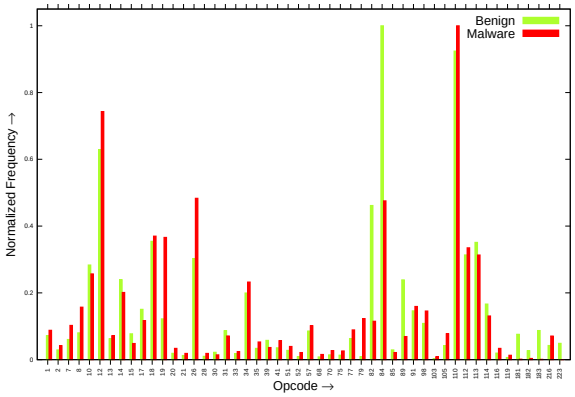


Figure 9: Top 50 opcodes occurrence difference between benign and malicious apps in the Storage group

Hence, the final features are selected after ordering the opcodes by their occurrence difference in each group (Al-

Table 2: Number of benign and Android malicious apps used for training and testing the classifiers

Groups	Train malware	Train benign	Test malware	Test benign	Total No. of apps
Calendar	59	57	14	14	144
Camera	179	423	44	106	752
Contacts	1073	356	268	89	1786
Location	1538	68	383	18	2007
Microphone	95	218	23	55	391
Others	110	891	27	223	1251
Phone	3981	1453	986	373	6793
SMS	2712	239	677	60	3688
Storage	2923	837	730	210	4700

gorithm 1) and used it for the detection of Android malicious apps.

Algorithm 1 : Feature Selection

INPUT: Pre-processed data
$\mathbf{N_B}$: No. of benign apps, $\mathbf{N_M}$: No. of malicious apps,
n: Total number of features required.
OUTPUT: List of features

BEGIN
for all benign and malicious apps **do**
Find the sum of frequencies $\mathbf{f_i}$ of each opcode **Op** and normalize it.

$$F_B(Op_j) = (\sum f_i(Op_j))/N_B$$

$$F_M(Op_j) = (\sum f_i(Op_j))/N_M$$

end for
for all opcode $\mathbf{Op_j}$ **do**

$$D(Op_j) = |F_B(Op_j) - F_M(Op_j)|$$

end for
return **n** number of prominent opcodes as features with high **D**(**Op**).

5 Classification of Malicious Apps

Ashu *et al.* [22] without grouping the data nor talking the apps permission investigated the top five classifiers viz. FT, RF, LMT, NBT and J48 for the classification of apps and reported that the FT is the best classifier and can detect the malicious apps with 79.27% accuracy [22]. Hence to improve the detection accuracy in this paper, first we grouped the apps based on the permissions and then classify the malicious apps using prominent opcode as the features (Figure 10). For the classification, the detail distribution (No. of training and testing malicious/benign apps, No. of apps in the group used for the classification) of the total collected dataset is given in Table 2. For the group-wise classification, we have used Waikato Environment for Knowledge Analysis (WEKA).

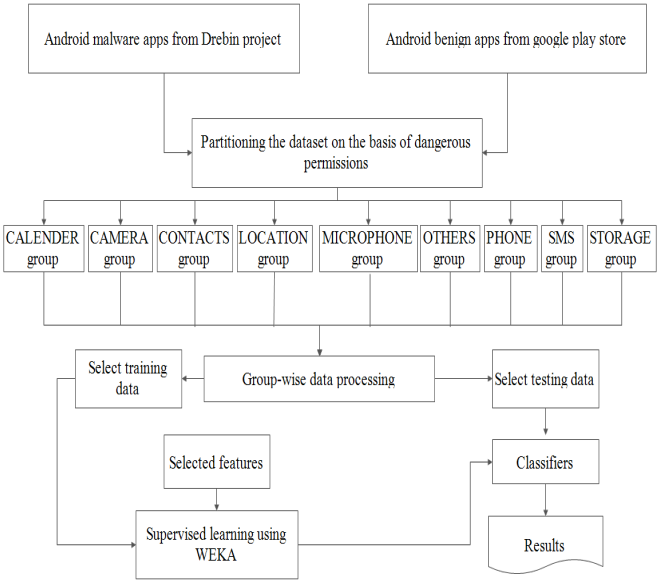


Figure 10: Flow chart for the detection of Android malicious apps by grouping the data

On the basis of studies [17, 20], we selected the same classifier (FT, RF, LMT, NBT, and J48) for the classification, but prominent features, training, and testing data are taken from the formed group only (Table 2). To measure the goodness of trained models, we evaluate the detection accuracy given by the equation

$$\text{Accuracy}(\%) = \frac{\text{True Positive} + \text{True Negative}}{\text{Total No. of Android Apps}} \times 100.$$

Where True Positive/Negative is the Android malicious/benign apps correctly classified [22].

The performance of the classifier has been investigated for each group by taking randomly 20% of the collected data (other than the training) with 20 - 200 best features incrementing 20 features at each step and the result obtained are shown in Figures 11 - 19 for the *Calendar, Camera, Contacts, Location, Microphone, Others, Phone, SMS,* and *Storage* group respectively.

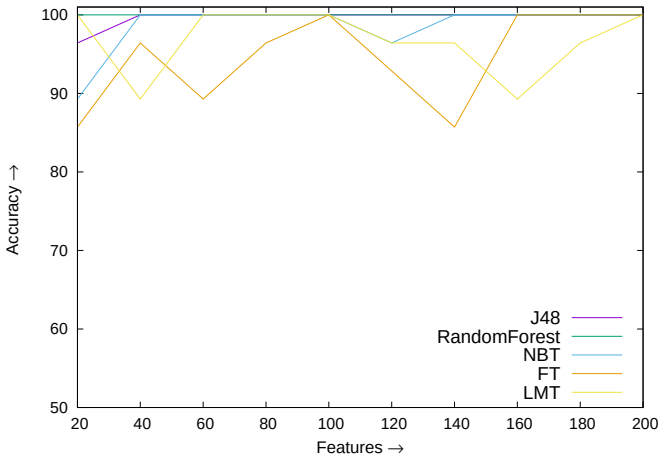


Figure 11: Detection accuracy obtained by the selected five classifiers for the Calendar group

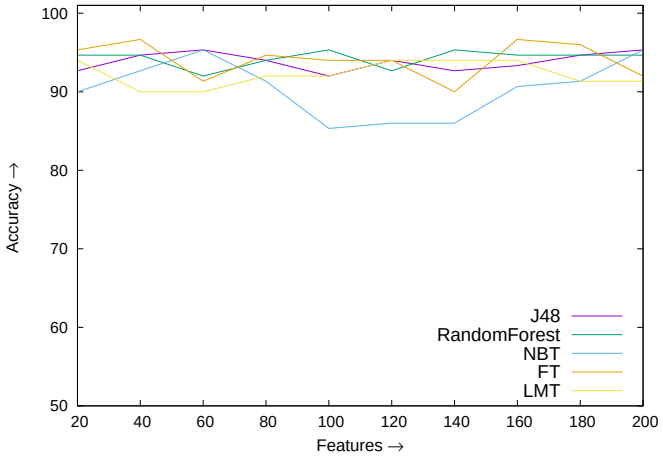


Figure 12: Detection accuracy obtained by the selected five classifiers for the Camera group

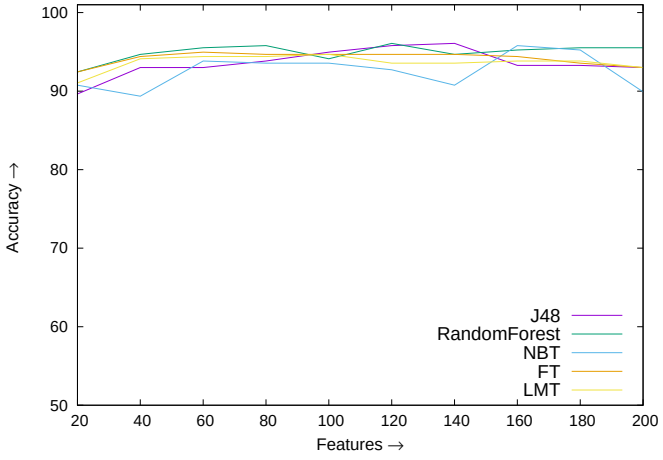


Figure 13: Detection accuracy obtained by the selected five classifiers for the Contacts group

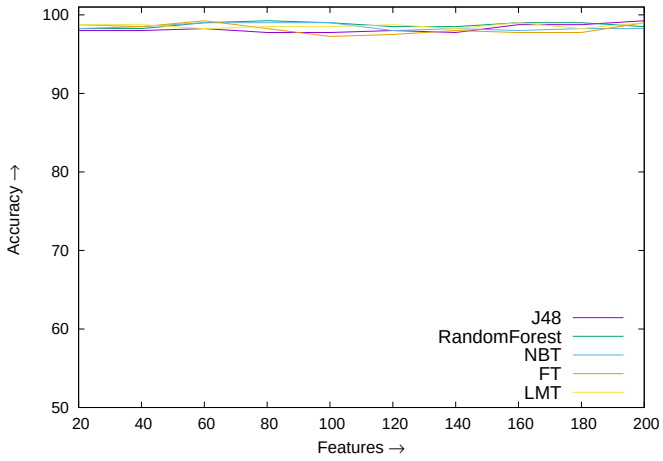


Figure 14: Detection accuracy obtained by the selected five classifiers for the Location group

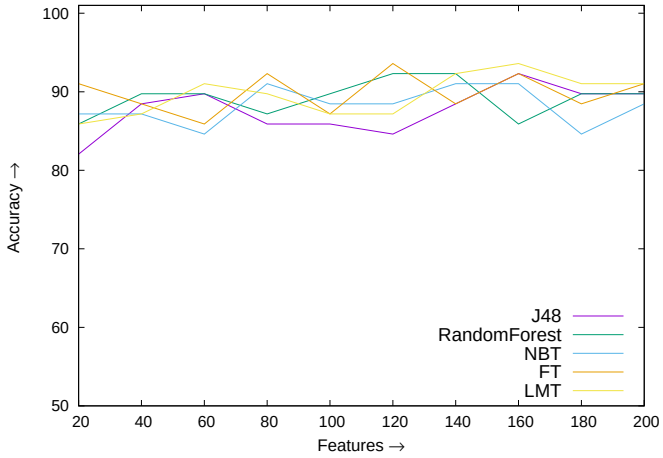


Figure 15: Detection accuracy obtained by the selected five classifiers for the Microphone group

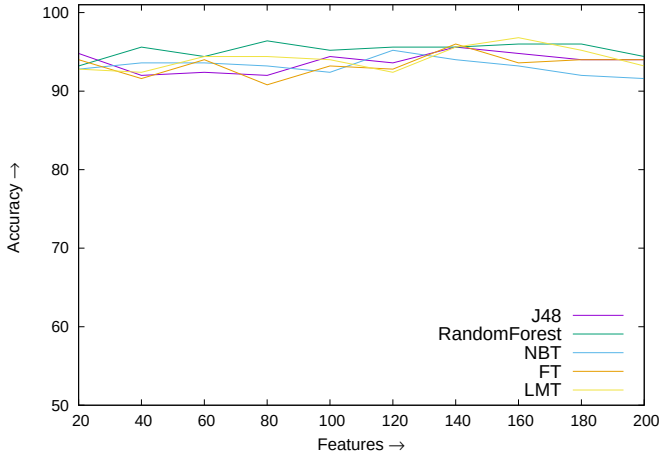


Figure 16: Detection accuracy obtained by the selected five classifiers for the Others group

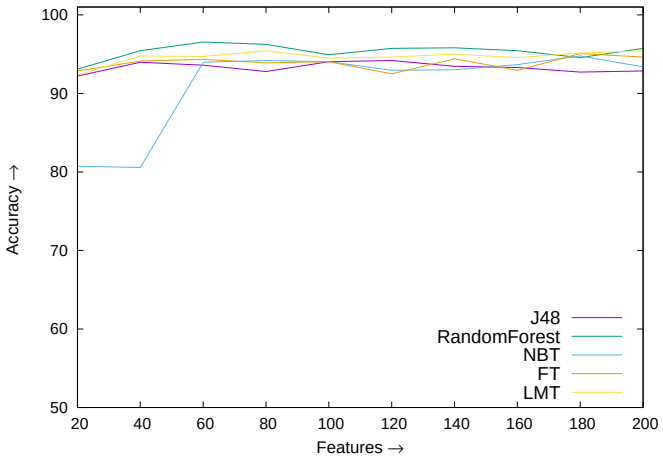


Figure 17: Detection accuracy obtained by the selected five classifiers for the Phone group

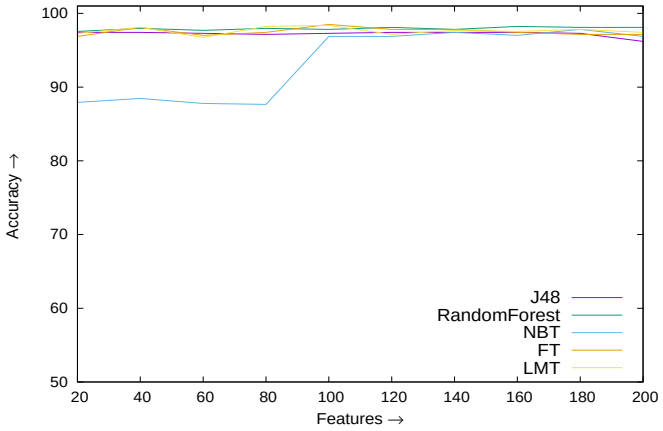


Figure 18: Detection accuracy obtained by the selected five classifiers for the SMS group

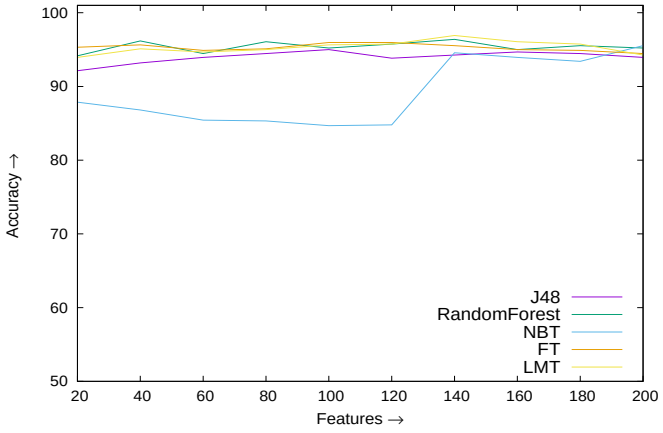


Figure 19: Detection accuracy obtained by the selected five classifiers for the Storage group

Table 3: Average accuracy obtained by the five classifiers

No. of Features	J48	RF	NBT	FT	LMT
20	93.69	95.01	90.37	93.32	94.28
40	95.28	96.26	92.26	93.78	93.45
60	95.51	96.10	94.24	94.01	94.31
80	94.83	**96.32**	94.44	95.38	95.46
100	**95.15**	96.24	94.41	**95.43**	**85.47**
120	94.48	95.96	92.96	94.57	94.23
140	95.12	96.08	93.68	93.53	94.76
160	95.39	95.16	**94.97**	95.16	94.29
180	94.94	95.73	93.93	95.18	94.56
200	94.71	95.78	93.24	94.98	94.71
Maximum	**95.51**	**96.32**	**94.97**	**95.43**	**95.47**
Minimum	93.69	95.01	90.37	93.32	93.45

The average accuracy obtained by the selected classifier are shown in Table 3. Here, the average accuracy means the sum of accuracy obtained by the classifier in the individual group with a fixed number of features divided by the total number of groups.

The analysis shows that RF average detection accuracy is best among the five classifiers and fluctuates least with the number of features, whereas NBT performance is worst and fluctuate maximum with the number of features.

However, the maximum average accuracy obtained by the selected five classifiers does not fluctuate much (94.97% - 96.32%) but minimum average accuracy fluctuation is high (90.37% - 95.01%), and for the best performance one shall take top 80 - 100 features, for the training and testing. The best accuracy obtained by the classifier in all the groups are given in Table 4.

We find that the detection accuracy is maximum in the Calendar group and minimum in the Microphone group obtained by FT and RF classifier respectively. The overall average maximum accuracy comes to 97.15%, which is very much better than then the obtained accuracy without grouping and taking permissions into account [22] and Arp, *et al.* (94%, 2014), Annamalai *et al.* (84.29%, 2016), Bahman Rashidi *et al.* (82%, 2017), Ali Feizollah, *et al.* (95.5%, 2017) (Figure 20).

In terms of TP i.e. detection rate of malicious apps, the *Calendar* group are best classified by RF and *SMS* group are least by FT, while in terms of TN i.e. benign detection rate, *Calendar*, and *SMS* group are best classified with RF and FT classifier respectively, while Others group containing normal permissions is best classified by the LMT classifier. The group-wise results of TP and TN obtained by the classifiers which give the best accuracy are shown in Table 4.

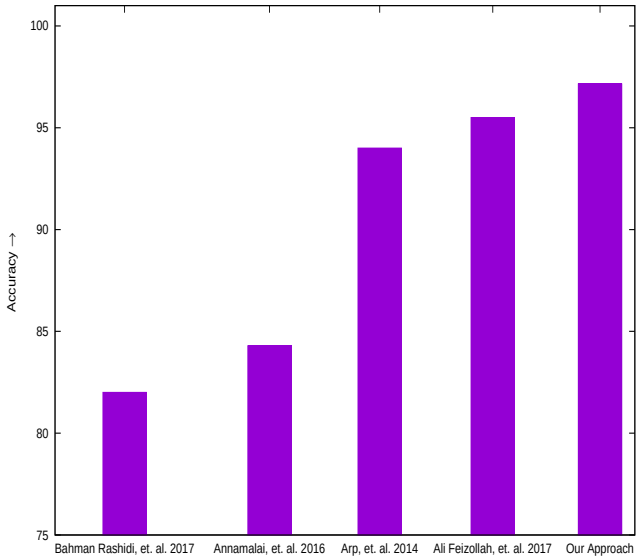


Figure 20: Comparisons of accuracy achieved by us with four other authors

Table 4: Group-wise maximum accuracy, TP and TN obtained by the classifiers

Groups	Best Classifier	Accuracy	Features Required	TN	TP
Calendar	RF	100.00	20	1.00	1.00
Camera	FT	96.67	40	0.93	0.98
Contacts	RF	96.08	120	0.99	0.89
Location	FT	99.25	60	0.99	0.94
Microphone	FT	93.59	120	0.87	0.96
Others	LMT	96.80	160	0.85	0.98
Phone	RF	96.54	60	0.98	0.92
SMS	FT	98.51	100	1.00	0.80
Storage	LMT	96.91	140	0.99	0.88

6 Conclusion

For the smart devices users, millions of Android apps are available at Google Play store and by the third party. Some of these available apps may be malicious. To defend the threat/attack from these malicious apps, a timely counter-measures has to be developed. Therefore, in this paper using *Drebin* benchmark malware dataset we group-wise analyzed the collected data based on permissions and experimentally demonstrated how to improve the detection accuracy of Android malicious apps and achieved 97.15% average accuracy. The obtained results outperformed the accuracy achieved by without grouping the data (79.27%, 2016), Arp, *et al.* (94%, 2014), Annamalai *et al.* (84.29%, 2016), Bahman Rashidi *et al.* (82%, 2017)) and Ali Feizollah, *et al.* (95.5%, 2017). The outperformance of our approach with the compared author results is basically due to the use of logic of the apps resides in the *.smalli* file and developing nine different models for the classification. Among the groups, the *Microphone* group detection accuracy is least while *Calendar* group apps are detected with maximum accuracy and for the best performance, one shall take top 80 - 100 features. In term of TP i.e. detection rate of malicious apps, *Calendar* group is best classified by RF, and *SMS* group is least by FT, while in terms TN i.e. benign detection rate, *Calendar*, and *SMS* group are best classified by RF and FT classifier respectively, while Others group containing normal permissions is best classified by the LMT classifier. It appears that group-wise detection of Android malicious apps will be efficient than without grouping the data. Hence, for the efficient classification of apps, in-depth study is required to optimize the feature selection, identifying the best-suited classifier for the group-by-group analysis. In this direction, work is in progress and will be reported elsewhere.

Acknowledgements

Mr. Ashu Sharma is thankful to BITS, Pilani, K.K. Birla Goa Campus for the Ph.D. scholarship No. Ph603226/July 2012/01. We are also thankful to Technische Universitat Braunschweig for providing the Drebin dataset for research on Android malware.

References

[1] Android-developers, *Normal and Dangerous Permissions requesting permissions*, Technical Report, Android Labs, 2017.

[2] D. Arp, M. Spreitzenbarth, M. Hubner, H. Gascon, and K. Rieck, "Drebin: Effective and explainable detection of android malware in your pocket," in *NDSS*, pp. 1–15, 2014.

[3] G. Canfora, F. Mercaldo, and C. A. Visaggio, "An HMM and structural entropy based detector for android malware: An empirical study," *Computers & Security*, vol. 61, pp. 1–18, 2016.

[4] S. Das, Y. Liu, W. Zhang, and M. Chandramohan, "Semantics-based online malware detection: Towards efficient real-time protection against malware," *IEEE Transactions on Information Forensics and Security*, vol. 11, no. 2, pp. 289–302, 2016.

[5] A. Feizollah, N. B. Anuar, R. Salleh, S. T. Guillermo, and S. Furnell, "Androdialysis: Analysis of android intent effectiveness in malware detection," *Computers & Security*, vol. 65, pp. 121–134, 2017.

[6] H. Gonzalez, N. Stakhanova, and A. A. Ghorbani, "Droidkin: Lightweight detection of android apps similarity," in *International Conference on Security and Privacy in Communication Systems*, pp. 436–453, 2014.

[7] J. Jang, H. Kang, J. Woo, A. Mohaisen, and H. K. Kim, "Andro-dumpsys: Anti-malware system based on the similarity of malware creator and malware centric information," *computers & security*, vol. 58, pp. 125–138, 2016.

[8] Q. Jerome, K. Allix, R. State, and T. Engel, "Using opcode-sequences to detect malicious android applications," in *2014 IEEE International Conference on Communications (ICC'14)*, pp. 914–919, 2014.

[9] McAfee, *Mcafee Labs Threats Report*, Dec. 2016. (https://www.mcafee.com/enterprise/en-us/assets/reports/rp-quarterly-threats-dec-2016.pdf)

[10] N. Milosevic, A. Dehghantanha, and K. K. R. Choo, "Machine learning aided android malware classification," *Computers & Electrical Engineering*, 2017.

[11] A. Narayanan, L. Yang, L. Chen, and L. Jinliang, "Adaptive and scalable android malware detection through online learning," in *International Joint Conference on Neural Networks (IJCNN'16)*, pp. 2484–2491, 2016.

[12] S. Naval, V. Laxmi, M. Rajarajan, M. S. Gaur, and M. Conti, "Employing program semantics for malware detection," *IEEE Transactions on Information Forensics and Security*, vol. 10, no. 12, pp. 2591–2604, 2015.

[13] K. Olmstead and M. Atkinson, *Apps Permissions in the Google Play Store*, Pew Research Center, 2015. (http://www.pewinternet.org/2015/11/10/apps-permissions-in-the-google-play-store/)

[14] G. Paller, *Dalvik Opcodes*, Android labs, 2017. (http://pallergabor.uw.hu/androidblog/dalvik_opcodes.html)

[15] QuickHeal, *Threat Report 3rd Quarter*, Technical Report, Quick Heal, 2015.

[16] B. Rashidi, C. Fung, and E. Bertino, "Android resource usage risk assessment using hidden markov model and online learning," *Computers & Security*, vol. 65, pp. 90–107, 2017.

[17] S. K. Sahay and A. Sharma, "Grouping the executables to detect malwares with high accuracy," *Procedia Computer Science*, vol. 78, pp. 667–674, 2016.

[18] A. Saracino, D. Sgandurra, G. Dini, and F. Martinelli, "Madam: Effective and efficient behavior-based android malware detection and prevention," *IEEE Transactions on Dependable and Secure Computing*, vol. 99, pp. 1–14, 2016.

[19] S. H. Seo, A. Gupta, A. M. Sallam, E. Bertino, and K. Yim, "Detecting mobile malware threats to homeland security through static analysis," *Journal of Network and Computer Applications*, vol. 38, pp. 43–53, 2014.

[20] A. Sharma, S. K. Sahay, and A. Kumar, "Improving the detection accuracy of unknown malware by partitioning the executables in groups," in *Advanced Computing and Communication Technologies*, pp. 421–431, 2016.

[21] A. Sharma and S. K. Sahay, "Evolution and detection of polymorphic and metamorphic malwares: A survey," *International Journal of Computer Applications*, vol. 90, pp. 7–11, Mar. 2014.

[22] A. Sharma and S. K. Sahay, "An investigation of the classifiers to detect android malicious apps," in *Information and Communication Technology*, pp. 207–217, 2017.

[23] A. Shaun, A. Tareq, C. Peter, C. Mayee, and D. Jon, *Internet Security Threat Report 2017*, Symentec, 2017. (https://www.symantec.com/content/dam/symantec/docs/reports/istr-22-2017-en.pdf)

[24] Statista, *Number of Available Applications in the Google Play Store from December 2009 to February 2016*, Technical Report, Statista, Aug. 2016.

[25] Symentec, *Internet Security Threat Report 2016*, Symantec Corporation, 2016. (http://or.himsschapter.org/sites/himsschapter/files/ChapterContent/or/FinnD_ORHIMSS_Spring16_Conf.pdf)

Biography

Mr. Ashu Sharma was born in Jhansi, Uttar Pradesh, India. He received his Bachelor's degree in Computer Science and Engineering from Uttar Pradesh Technical University and Master's degree in Information Security from Atal Bihari Vajpayee Indian Institute of Information Technology and Management, Gwalior. In 2012 he joined the Department of Computer Science and Information Systems, BITS, Pilani, K.K. Birla Goa Campus, India as a full-time research scholar for the Ph.D. degree under the supervision of Dr. Sanjay K. Sahay. He has published several papers in reputed journals and conferences.

Dr. Sanjay Kumar Sahay is working as an Assoicate Professor in the Department of Computer Science and Information Systems, BITS, Pilani, K.K. Birla Goa Campus. He is also a Visiting Associate of IUCAA, Pune. His research interests are Information Security, Data Science, and Gravitational Waves. He basically teaches Network Security, Cryptography, Computer Networks, and Data Mining courses. Before joining BITS, Pilani, and after submitting his Ph.D. thesis on "Studies in Gravitational Wave Data Analysis" during 2002-2003, he continued his work on Data Analysis of Gravitational Waves as a Project Scientist at IUCAA, Pune, India. In 2003-2005 at Raman Research Institute, Bangalore, India he worked as Project Associate on the multi-wavelength astronomy project (ASTROSAT), where he worked on the data pipeline of Scanning Sky Monitor. In 2005 he worked as Post Doctoral Fellow at Tel Aviv University.

Multipath Key Exchange Scheme Based on the Diffie-Hellman Protocol and the Shamir Threshold

Daouda Ahmat[1], Marayi Choroma[2], and Tegawendé F. Bissyandé[3,4]
(Corresponding author: Daouda Ahmat - daouda.ahmat@uvt.td)

Virtual University of Chad[1]
B.P: 5711, General Daoud Soumaine Road, N'Djamena, Chad
University of N'Djamena[2]
B.P: 1117, Mobutu Street, N'Djamena, Chad
University of Luxembourg, SnT[3]
2, avenue de l'Université L-4365 Esch-sur-Alzette, Luxembourg
University Ouaga I Pr Joseph Ki-Zerbo[4]
B.P: 7021, Ouagadougou, Burkina Faso
(Email: daouda.ahmat@uvt.td[1] / choroma.marayi@auf.org[2] / tegawende.bissyande@uni.lu[3,4])
(Received Dec. 18, 2017; Revised and Accepted June 18, 2018; First Online Jan. 13, 2019)

Abstract

In the Internet, as well as in any open autonomous distributed systems, threats to secure communications are pervasive. We contribute towards adressing them by proposing, in this paper, a new multipath key exchange approach, which does not rely on any centrally trusted coordinator. This approach is thus suitable for use in distributed systems such as widespread P2P networks or booming wireless mesh networks (e.g., for the Internet-of-Things). We design a new algorithm based upon an extension of both the Diffie-Hellman protocol and the Shamir threshold scheme. In order to overcome man-in-the-middle attacks inherent to the Diffie-Hellman key exchange model, our proposed approach guarantees secure key exchange by exploring disjoint transmission paths and the Shamir threshold scheme. The public key is used as the root of a polynomial of degree $k-1$, and n points of this polynomial are generated and transmitted from source to destination, each point through a disjoint path. Upon reception of at least k points among n, the receiver is able to reconstruct the complete key. In addition, this paper demonstrates how the disjoint paths constructions and the routing algorithms are designed to work regardless of the network topology.

Keywords: Diffie-Hellman; Key Exchange; Multipath Routing; P2P Networks; Shamir Secret Sharing

1 Introduction

A growing number of security attacks on distributed systems such as the Internet and P2P networks for the Internet-of-things (IoT), have led to an increasing interest in the research community. In this regard, many solutions have been proposed to secure traffic across networks by using a security infrastructure based on a central authority through which s cryptographic keys are dispatched. However, centralized key exchange systems are not suitable for autonomous distributed systems such as peer-to-peer networks. Infrastructure-less key exchange techniques, which are not tied to a central node for key negotiation, and which can be usable over *insecure* networks are thus necessary in the context of such systems.

The end-to-end key exchange scheme proposed by the Diffie-Hellman protocol [11] enables a key exchange between two remote correspondents. However, due to its vulnerability to interception attacks, the Diffie-Hellman protocol cannot be leveraged as-is. Our approach, in this paper, aims at overcoming this issue by combining the Diffie-Hellman protocol with both multipath routing and the Shamir's threshold scheme [32]. Concretely, the Shamir's secret sharing enables us to divide a key $\mathcal{K}$ into n subkeys in such a way that $\mathcal{K}$ is reconstructable from any k subkeys. A security property in this scheme is that any subset of up to $k-1$ subkeys cannot leak information about $\mathcal{K}$.

The contributions of the paper are exposed in the re-

mainder of the paper, which is structure as follows:

- We first discuss the scope and features provided by literature work, then we enumerate the limitations of our own previous work towards implementing secure network exchanges (cf. Section 2).
- We then present background information related to previous work (cf. Section 3), before providing detailed descriptions on the design of our key exchange scheme (cf. Section 4), including the implementation of several multipath routing approaches (cf. Section 5).
- Finally, we analyse the security advantages of our approach (cf. Section 6) and discuss experimental results —based on network simulations— that assess the efficiency of the proposed approach (cf. Section 7).

2 Related Work

A number of research works have presented various security infrastructures over fully decentralized or ad hoc networks [5,10,13,23,35,41]. Although they are designed to be suitable in such environments, the proposed approaches come with different caveats. In this section, we describe some models from the literature and highlight the potential benefits of our approach.

Srivasta and Liu have relied on the Diffie-Hellman algorithm to deliver a solution that prevents threats in DHT networks [34]. Wang *et al.* have built a distributed PKI on top of the Chord structured overlay network [2]. They have used threshold cryptography to distribute the functionality of the PKI across the nodes of the DHT network. This Chord-PKI provides traditional PKI features such as certification, revocation, storage and retrieval.

The literature now includes a number of approaches [8, 9,12,15,17,22,24,26–30,40] that extend the Diffie-Hellman key exchange algorithm. Nevertheless, there are scarce works which address end-to-end key exchange problem based on both distributed systems such as peer-to-peer networks by leveraging the Diffie-Hellman protocol and proposing multipath subkeys routing and multi-secret mechanisms. Takano *et al.* [35] have investigated this avenue over a decade ago. This approach is based on ring topology and does not provide explanation about its key splitting technique.

Jiejun *et al.* propose to distribute certification authority functions through a threshold secret sharing mechanism [21]. In this system, the private key is computed by k neighbor nodes and the public key is derived from node identity.

Many threshold schemes have been directly derived from traditional Shamir threshold [32] to address multi-secret sharing mechanisms. Among them we can quickly cite Brian King [20], Appala *et al.* [36], Harsha *et al.* [19], Rao *et al.* [39], Yang *et al.* [42], Guo *et al.* [18], Ting-Yi Chang *et al.* [38], Ting-Yi Chang [7] and Chang *et al.* [6]. We will focus our study on the mainstream protocol known under name of the *Shamir threshold* [32].

Fathimal *et al.* [16] recently proposed an extension of Shamir's method that enables to retrieve a key from p subkeys, where $p \leq k-1$. p is a threshold number of equally-weighted from each compartment.

Threshold cryptography is also used in identity-based key management [10]. The main idea for identity-based cryptography is to define public keys derived from the identities of communicating nodes [33]. Unfortunately, node identity updates lead to frequent key changes.

Myrmic [41] is a DHT-based system that proposes a secure and robust routing protocol. Designed to be robust against adversarial interference, Myrmic introduces the concept of *Neighborhood Authority* in order to handle certificates in a small set of nodes.

Takano *et al.* have designed a Multipath Key Exchange [35] similar to that proposed in our work. Their technique however was designed to fit the Symphony and Chord P2P systems that are both based upon a ring topology. Their proposed approach, based on probabilistic clockwise/anticlockwise routing, is thus sensitive to coordinated MITM attacks by two attackers.

Jaydip Sen proposes a multipath certification protocol for MANETs that proceeds by broadcasting in order to discover the route between both source and destination nodes [31]. The key exchange protocol is based on this routing approach to retrieve the public keys of the nodes. However, broadcasting techniques have proven to not be relevant for large scale networks such as fully decentralized P2P systems.

El Hajj Shehadeh *et al.* investigate secret key generation from wireless multipath channels [13]. The proposed protocol is based mainly on both the physical characteristics of the wireless channel and a key pre-distribution scheme. This solution is implemented within the physical layer and does not scale to large networks.

3 Background

In previous work [1], we have proposed a key exchange scheme based on an extension of the Diffie-Hellman protocol. Our approach enabled sharing a secret by using multiple disjoint paths in a P2P system called CLOAK [4,37]. The scheme was mainly devised to overcome the vulnerability of the Diffie-Hellman protocol to Man-In-The-Middle (MITM) attacks.

In this scheme, we would split the public key into several subkeys that would then be sent over several disjoint paths to the destination. The destination needed to recover all subkeys in order to get the public key of the sender. A major issue with this approach was that the interception of most (*i.e.*, not even all) of the subkeys by an attacker, allows him to make a brute force attack on the missing subkey(s) which are smaller than the original key. Indeed, if the subkeys set is $S^K = \{0,1\}^*$, where $|S^K| = \rho$, interception of α key components among n by

an attacker, with $\alpha \leq n$, reduces the difficulty to carry out a brute force attack to $\frac{\rho-\alpha\frac{\rho}{n}}{n}$, that is $\rho\frac{n-\alpha}{n^2}$. In addition, in our previous work, the proposed multipath routing algorithm for the subkeys was exclusively suitable to our CLOAK P2P system [37], making its exploitation challenging in other systems.

In this work, we try to address the issues and limitations of this prior work by integrating the Shamir's shared secret scheme in our previous solution.

4 Scheme Design

In this section, we describe our key exchange scheme and its corollary properties which are suitable to distributed networks, *i.e.*, networks that lack any trusted central coordination point.

4.1 Diffie-Hellman Vulnerability

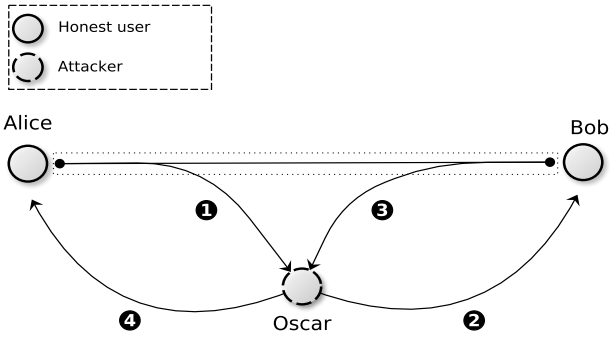


Figure 1: Man-in-the-middle attack

The Diffie-Hellman protocol is an algorithm initiated by two distant correspondents that cooperate to remotely accomplish key exchange tasks. As shown in Figure 1, one fundamental problem of the Diffie-Hellman protocol is its vulnerability to interception attacks, known as Man-In-The-Middle (MITM) attacks. This figure displays a scenario where an attacker, *Oscar*, eavesdrops the channel used by both *Alice* and *Bob* to exchange cryptographic data. ❶ and ❸ flows represent intercepted data by the attacker *Oscar*, data transmitted respectively by *Alice* and *Bob*. ❷ and ❹ show corrupted flows produced by *Oscar* and then respectively transmitted to *Bob* and *Alice*. Thus, *Oscar* can perpetrate attacks: it can eavesdrop, replay or modify data exchanged between *Alice* and?*Bob*. Specifically to the Diffie-Hellman protocol, *Oscar* can intercept the public key sent by *Alice* and send its own public key to *Bob* and can do the same in the other direction with another public key generated to replace the public key sent by *Bob*.

4.2 Protocol Overview

Based upon multipath routing, our approach aims at reducing the Diffie-Hellman vulnerability by combining the two cryptographic algorithms mentioned earlier. Hence, key $\mathcal{K} = g^s \bmod (p)$ must be splitted into n subkeys $s_{k_0}, \ldots, s_{k_{n-1}}$ and then each subkey s_{k_i} will be subsequently sent through a disjoint path. Shamir's threshold algorithm is applied in both splitting and reconstruction of key $\mathcal{K}$. In addition, various routing techniques are proposed in order to route subkeys over network through disjoint paths.

4.3 Key Management

Our key exchange approach is based upon both the Diffie-Hellman protocol and Shamir's threshold. We now describe the mechanisms behind our key exchange scheme.

4.3.1 Key Splitting

In order to forge subkeys, each correspondent firstly creates a secret key $S^K = s$ and generates a polynom $f^l(X)$, where $f^l(0) = g^s (\bmod\ p)$, as shown in Algorithm 1. Then, for each $x_{i\ 0\leq i\leq n}$, with $x_i \neq 0$, $f^l(x_i)$ is computed. Finally, all interpolation points $(x_i, f^l(x_i))$, except $(0, f^l(0))$, are stocked in `subKeysList`. Algorithm 2, which depends on Algorithm 1, provides more details about the key splitting scheme.

Algorithm 1: Creation of polynom of degree k

`createPolynom`(k, $g^S (\bmod\ p)$) **return** Polynom;
begin
  $a_0 \leftarrow g^S (\bmod\ p)$;
  $f^l(0) \leftarrow a_0$;
  $i \leftarrow 1$;
  while $i \leq k-1$ **do**
    $a_i \leftarrow$ `getRandomCoefficient`();
    if $i = k-1$ and $a_i = 0$ **then**
      continue;
    $l_i(X) \leftarrow a_i X^i$;
    $i \leftarrow i+1$;
  $f^l(X) \leftarrow \sum_{i=1}^{k-1} l_i(X) + f^l(0)$;
  return $f^l(X)$;

4.3.2 Key Reconstitution

On receiving of $(x_i, f^l(x_i))_{0\leq i\leq n}$, correspondent node applies Algorithm 3 in order to rebuild key $\mathcal{K} = g^s (\bmod\ p)$ from received subkeys $s_{k_0}, \ldots, s_{k_{p-1}}$, that are equivalent to $(x_i, f^l(x_i))_{0\leq i\leq p}$, where $p \geq k (=$ `degree of` $f^l)$. Indeed, receiver computes $s_{k_0} \odot s_{k_1} \odot \cdots \odot s_{k_{p-1}} = g^s (\bmod\ p) = \mathcal{F}^l(0)$, such that:

Algorithm 2: Subkeys generation

Input: $k, n,\ g^S(\text{mod } p)$
$f^l(X) \leftarrow \texttt{createPolynom}\left(k,\ g^S(\text{mod } p)\right)$;
subKeysList $\leftarrow \perp$;
begin
  $i \leftarrow 1$;
  while $i \leq n$ **do**
    $x_i \leftarrow \texttt{getRandomValue}()$;
    $\widehat{f}(x_i) \leftarrow \left(x_i, f^l(x_i)\right)$;
    $\texttt{storeInSubKeysList}\left(\widehat{f}(x_i), \text{subKeysList}[i]\right)$;
    $i \leftarrow i + 1$;

$$\mathcal{F}^l(X) = \sum_{j=0}^{p} y_j l_j(X) \tag{1}$$

where $y_i = f^l(x_i)$ and

$$l_j(X) = \prod_{i=0, i\neq j}^{p} \frac{X - x_i}{x_j - x_i} \tag{2}$$

Algorithm 3: Reconstitution of key from received subkeys

Input: $k, (x_i, y_i)_{0\leq i\leq n}$
Output: Key
begin
  if $|(x_i, y_i)_{0\leq i\leq n}| < k$ **then**
    return $\perp$;
  else
    foreach $i \in [\![0, n]\!]$ **do**
      $l_i(X) \leftarrow \prod_{j=0, j\neq i}^{n} \frac{X - x_j}{x_i - x_j}$;
    $j \leftarrow 0$;
    $f_l(X) \leftarrow 0$;
    while $j \leq n$ **do**
      $f_l(X) \leftarrow y_j \times l_j(X) + f_l(X)$;
      $j \leftarrow j + 1$;
    return $f_l(0)$;

4.4 Key Exchange Protocol

Algorithm 4 summarizes the process of our key exchange approach: the Diffie-Hellman protocol is relied upon firstly to generate a key of shape $\mathcal{K} = g^s(\text{mod } p)$; then Shamir's threshold is leveraged to split the key into several subkeys or to rebuild the key from its component subkeys $s_{k_0}, s_{k_1}, \cdots, s_{k_n}$. Precisely, equations 1 and 2 describe *Lagrange Interpolation* used in order to rebuild key $\mathcal{K}$ original.

Algorithm 4: Multipath key exchange protocol

public data:	**private data:**
$\boldsymbol{p}$: a prime number	$\mathbf{s}_a$: secret key of **Alice**
$\boldsymbol{g}$: a generator	$\mathbf{s}_b$: secrete key of **Bob**

1) **Alice** creates s_a and she then computes her partial key $\text{Key}_a = g^{s_a} \text{ mod } (p)$;
2) **Alice** generates a polynom f_a^l of degree k, such as $f_a^l(0) = \text{Key}_a$, and she then computes n interpolation points of the polynom: $\widehat{f}_{a_0}, ..., \widehat{f}_{a_{n-1}}$ (where $n \geq k$);
3) **Alice** sends n subkeys $\widehat{f}_{a_i}$, except $(0, f^l(0))$ point, to **Bob** via disjoint paths;
4) **Bob** determines $L^b(X)$ according to $\widehat{f}_{a_i}$, received from **Alice**, and subsequently computes $L^b(0)$ which gives $g^{s_a} \text{ mod } (p)$, if $|\widehat{f}_{a_i}| \geq k$;
5) **Bob** creates s_b and he then generates his partial $\text{Key}_b = g^{s_b} \text{ mod } (p)$;
6) **Bob** forges a polynom f_b^l, such as $f_b^l(0) = \text{Key}_b$, and he then determines n interpolation points of the polynom $\widehat{f}_{b_0}, ..., \widehat{f}_{b_{p-1}}$ (where $p \geq k$);
7) **Bob** sends n subkeys $\widehat{f}_{b_i}$, except $(0, f^l(0))$ point, to **Alice** through disjoint paths;
8) **Alice** generates $L^a(X)$ from $\widehat{f}_{b_i}$ received from **Bob** and she then computes $L^a(0)$ which gives $g^{s_b} \text{ mod } (p)$, if $|\widehat{f}_{b_i}| \geq k$;
9) **Alice** computes Key $= \text{Key}_a \times L(0) = g^{s_a} \text{ mod } (p) \times g^{s_b} \text{ mod } (p) = g^{s_a s_b} \text{ mod } (p)$;
10) **Bob** computes Key $= \text{Key}_b \times L(0) = g^{s_b} \text{ mod } (b) \times g^{s_a} \text{ mod } (p) = g^{s_b s_a} \text{ mod } (p)$;

5 Multipath Routing Policy

In this section, we provide technical details about multipath routing algorithms and then point out their performance differences.

5.1 Deterministic Routing: Pre-routing and Then Routing

Deterministic routing: Enables to route subkeys through disjoint and predetermined paths, as described in Algorithm 5. In other words, each subkey is sent via a disjoint path whose constituting hops are all determined in advance. *Deterministic routing* is however not suitable for dynamic environments where topologies change constantly.

5.2 Non-deterministic Routing: Both Marking and Routing

Non-deterministic routing: Detailed in Algorithm 6, enables to route each subkey through a disjoint path, but unlike deterministic routing, determines on the fly the hops that form each disjoint path.

Algorithm 5: Deterministic routing

Input: $G = (V, E), (s, t), \mathsf{SubKeysList}$
$\mathcal{V} \leftarrow \emptyset$;
begin
  $k \leftarrow |\mathsf{SubKeysList}|$;
  $j \leftarrow k - 1$;
  $\mathcal{V} \leftarrow \mathcal{V} \cup \{s, t\}$;
  while $j \geq 0$ **do**
    $node \leftarrow s$;
    while $node \neq t$ **do**
      $d_{min} \leftarrow \texttt{distance}(node, t)$;
      foreach $neighbor \in \texttt{neighborsListOf}(node)$ **do**
        $d \leftarrow \texttt{distance}(neighbor, t)$;
        if $neighbor \notin \mathcal{V}$ and $d < d_{min}$ **then**
          $d_{min} \leftarrow d$;
          $node \leftarrow neighbor$;
      $\mathcal{V} \leftarrow \mathcal{V} \cup \{node\}$;
      $\mathcal{P}_j \leftarrow \mathcal{P}_j \cup \{node\}$;
    $j \leftarrow j - 1$;
  while $k > 0$ **do**
    $e \leftarrow \mathsf{SubKeysList}[k]$;
    $\texttt{sendViaPath}(e, \mathcal{P}_k)$;
    $k \leftarrow k - 1$;

Algorithm 6: Non-deterministic routing

Input: $G = (V, E), (s, t), \mathsf{SubKeysList}$
$\mathcal{V} \leftarrow \emptyset$;
begin
  $\mathcal{V} \leftarrow \mathcal{V} \cup \{s, t\}$;
  foreach $e \in \mathsf{SubKeysList}$ **do**
    $node \leftarrow s$;
    while $node \neq t$ **do**
      $d_{min} \leftarrow (node, t)$;
      foreach $neighbor \in \texttt{neighborListOf}(node)$ **do**
        $d \leftarrow \texttt{distance}(neighbor, t)$;
        if $neighbor \notin \mathcal{V}$ and $d < d_{min}$ **then**
          $d_{min} \leftarrow d$;
          $node \leftarrow neighbor$;
      $\mathcal{V} \leftarrow \mathcal{V} \cup \{node\}$;
      $\texttt{forward}(e, node)$;

5.3 Technical Comparison Between Routing Algorithms

Table 1 presents a technical comparison of both performance metrics and features provided by various multipath routing algorithms.

6 Security Analysis

In a multipath key exchange scheme, a malicious node that wishes to compromise a key being exchanged must be able to collect each of all key components routed over the network. Formally, when paths $\mathcal{P}_0, ..., \mathcal{P}_{k-1}$ are used to send several distinct subkeys from source $\mathcal{S}$ to destination $\mathcal{D}$, the only malicious nodes that could compromise the key should be located at the intersection of all paths. In other words, all the malicious nodes belong to a set $M = \bigcap_{i=0}^{k} \mathcal{P}_i$ which represents the set of intersection points of all paths $\mathcal{P}_i$. $\mathcal{S}$ and $\mathcal{D}$ are obviously ignored in this set. Thus, when $\bigcap_{i=0}^{k} \mathcal{P}_i = \emptyset$ (*bigon criterion* is respected [14, Lemma 2.5]), then all paths are disjoint and any MITM attack attempt cannot succeed. In such a desirable case, there exists a k-connected subgrah between $\mathcal{S}$ and $\mathcal{D}$ in the network topology. When $|\bigcap_{i=0}^{k} \mathcal{P}_i| \geq 1$, there exists a real risk that MITM attacks could be committed on exchange transmitted between $\mathcal{S}$ and $\mathcal{D}$. That means that there exists at least one articulation point. Algorithm 7 enables to detect articulation points within network.

Consequently, the probability to have a MITM attack is estimated by $\sigma = \frac{|\bigcap_{i=0}^{k} \mathcal{P}_i|}{|\bigcup_{i=0}^{k} \mathcal{P}_i|}$ (where each path $\mathcal{P}_i$ is constituted of a set of consecutive hops from source $\mathcal{S}$ to destination $\mathcal{D}$). When all used paths are pairwise disjoint, the probability of *isolated* MITM attack (no *coordinated* MITM attack) is then: $\sigma = 0$ (i.e $|\bigcap_{i=0}^{k} \mathcal{P}_i| = 0$).

The number of distinct paths is also dependant on the source node's degree. Thus, for a given *q-regular tree*, if q is a large number, then there is a probability to have several disjoint transmission channels. Nonetheless, despite the robustness of our multipath negotiation approach, cooperative (*i.e.*, coordinated) MITM attacks, where several nodes maliciously cooperate to compromise a key, are possible. However, it is very hard, and excessively costly to launch such an attack in a real environment, especially in distributed systems where network topology changes dynamically. In addition, the key exchange scheme is suitable for P2P networks and designed regardless of a specific network architecture.

In order to improve performance, re-authentication feature is introduced. However, the challenge message used in this phase could be replayed. Furthermore, when a malicious node caches a challenge message, it can then create its copies and send them successively to target node. Thus, target node tries to resolves each challenge request because it does not know which packet is more fresh than the other. Consequently, it will be rapidly saturated with requests from malicious nodes. Therefore, this causes a Denial-of-Service (DoS) attack.

In order to avoid such an attack from malicious nodes,

Table 1: Comparison of Multipath routing strategies

Finding disjoint paths	Complexity and various features				
	Time complexity	Space complexity	Parity [a]	Robustness [b]	Overview [c]
Indeterministic routing	$O(k(\|E\| + \|V\| \log \|V\|))$	$O(k\|V\|)$		$\surd$	
Deterministic routing	$O(k(\|E\| + \|V\|(1 + \log \|V\|)))$	$O(k\|V\|)$	$\surd$		
Menger's theorem	NP	–	–		$\surd$

[a] Parity between the number of disjoint paths and the number of generated subkeys
[b] Resilience to topology change
[c] Knowledge of topology is needed

Table 2: technical comparison of key exchange schemes

Key Exchange Method	Diffie-Hellman	Takano *et al.*	Our model
Robust to MITM		$\surd$	$\surd$
Unpredictable paths	– [b]		$\surd$
Free of particular topology	$\surd$		$\surd$
CMITM[a] implementation	easy	easy	hard
Required subkeys among n	–	n	$k \leq n$
Subkeys robustness level	–	medium	high
|Set of disjoint paths|	$= 0$	> 2	> 2
Topology maintaining cost	$\mathcal{O}(0)$	$\mathcal{O}(log^2 n)$	$\mathcal{O}(0)$

[a] Coordinated MITM attacks
[b] It depends to the knowledge or not of network topology

a timestamp is assigned to each encrypted challenge message. Thus, the target node could distinguish between fresh packets and replayed packets.

Furthermore, during the key negotiation phase, all packets are exchanged in a clear text mode. Thus, traffic analysis attacks could reveal details about captured packets such as *sequence number* or *payload* which is nothing other than the transported subkey. Hence, multipath key exchange is needed to prevent the knowledge of all subkeys.

Table 2 summarizes security and technical features of traditional security protocol, called Diffie-Hellman algorithm [11], key exchange scheme proposed by Takano *et. al* [35] and our key management scheme. This table shows that our scheme is more advanced than other models in several aspects.

Isolated attacks launched over the network cannot compromise multipath key exchange if there are at least two disjoint paths found between two correspondent nodes. However, coordinated attacks launched from various malicious nodes could be potentially able to compromise key by intercepting all its subkeys sent through disjoint paths.

Otherwise, in the new scheme that is proposed in this paper, missing a few of the subkeys, during their transport, does not always cause key exchange failure. Technically, if the number of received keys is greater or equal to the threshold k, with $k \leq n$, then the original key could be reconstructed. Formally, the assertion can take the

Algorithm 7: Articulation point detection

```
getArticulationPoint(s, t) return node;
begin
    i ← j ← 0;
    neighborsList ← neighborsListOf(s);
    foreach neighbor ∈ neighborsList do
        P_j ← ∅;
        createPath(neighbor, t, P_j);
        node ← P_j.getLastNode();
        if node ∉ t then
            i ← i + 1;
        P_j ← P_j ∪ {node};
        j ← j + 1;
    return (i = j) ? ⋂_{i=0}^{n} P_i : ⊥;
```

form of the following theorem.

Theorem 1. *In Shamir's $(k,n)-$threshold scheme, any subset of up to $q = k-1$ subkeys, where $k \leq n$, does not leak any information on the shared secret $\mathcal{K}$.*

Proof. To retrieve key $\mathcal{K}$ from (x_i, y_i) which are employed in Equation (1), let's proceed as follow:

$$\mathcal{K} = \mathcal{F}^l(0) = \sum_{j=0}^{p} y_j \prod_{i=0, i\neq j}^{p} \frac{-x_i}{x_j - x_i} \tag{3}$$

where $k \leq p \leq n$.

Given $|(x_i, y_i)|_{1\leq i\leq q}$, with $q < k \Rightarrow q < p$. Let's suppose that $p = k$, that means that Equation (3) becomes in developped form:

$$\mathcal{K} = \sum_{j=0}^{q} y_j \prod_{\substack{i=0 \\ i\neq j}}^{q} \frac{-x_i}{x_j - x_i} + \sum_{j=q+1}^{k} y_j \prod_{\substack{i=q+1 \\ i\neq j}}^{k} \frac{-x_i}{x_j - x_i} \tag{4}$$

In Equation (4) let's put:

$$\sum_{j=0}^{q} y_j \prod_{\substack{i=0 \\ i\neq j}}^{q} \frac{-x_i}{x_j - x_i} = \mathcal{K}_0 \tag{5}$$

and

$$\sum_{j=q+1}^{k} y_j \prod_{\substack{i=q+1 \\ i\neq j}}^{k} \frac{-x_i}{x_j - x_i} = \mathcal{K}_1 \tag{6}$$

Therefore Equation (4) becomes:

$$\mathcal{K} = \mathcal{K}_0 + \mathcal{K}_1 \tag{7}$$

The value of $\mathcal{K}_1$ is indeterminate because points $(x_i, y_i)_{q+1\leq i\leq k}$ are unknown. That implies $\mathcal{K}$ indeterminate. □

The public key cryptography $g^{s_{k_i}}$ can be published and used in order to verify authenticity of each subkey s_{k_i}. However, public key mechanism requires the use of a traditional centralized public key infrastructure that is incongruous to distributed systems such as peer-to-peer networks.

7 Protocol Assessment

We rely on the Erdõs-Rényi and the Magoni-Pansiot [25] models to build a synthesized graph that represents a random topology. To assess our approach, we use the *nem* simulator[1].

[1] http://www.labri.fr/perso/magoni/nem/

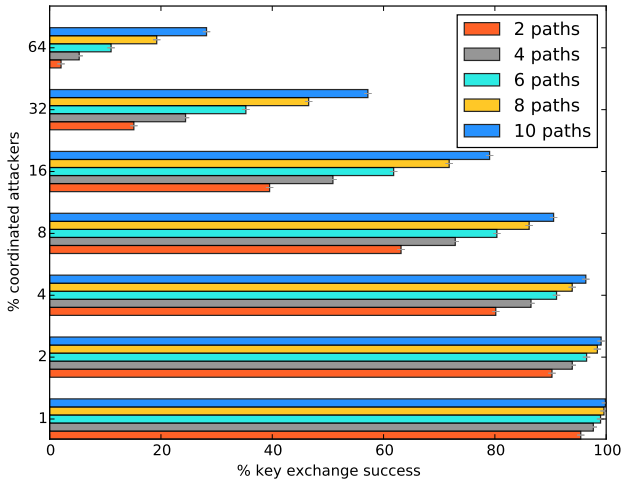


Figure 2: Average number for 10 assessment rounds of key exchange success with respect to both various numbers of disjoint paths found and percentages of coordinated attackers existing over the network

We have carried out the experiments through simulations. We succinctly present the steps that are carried out for the experiments:

- *Definition of the network:* A P2P network is first created. In this step, we set the type of the topology (real map, synthesized topology such as Erdős-Reńyi, Internet-like, etc.) and the size of this network.
- *Selection of the set of compromised nodes:* In the second step, we select a subset of X% nodes which will act as attackers. These nodes are supposed to coordinate their actions.
- *Identification of source and destination nodes:* We then select, among the non-compromised nodes, a pair of source and destination nodes for the data exchange.
- *Data packet transfer:* We launch the transmission by transferring a data packet through the shortest path towards the destination node. All intermediate nodes will be marked and may not be used for another packet between the same pair of source/destination nodes.
- *Check for attacks:* At the end of the packet transfer, we check whether the packet was intercepted by an attacker.
- *Use of alternative paths:* At this time, we start over from step 4, using the same source node but a different path to reach the same destination.
- *Confirmation of the validity of the generated key:* We check whether the packet was potentially intercepted on all disjoint paths. If this is the case, then this

attempt to generate a key is a failure. It is a success otherwise.

- *Change of source/destination nodes:* We repeat the experiments starting from step 3 with a new pair of source and destination nodes.
- *Change of compromised nodes set:* We repeat the experiments starting from step 2 with a new subset of compromised nodes. Basically, we change the percentage of attackers.
- *Change of network settings:* We start over the experiments from step 1 with a new network topology and/or a new size value for the network.

Assessment results are depicted in Figure 2. On the one hand, and despite coordinated attacks, the results show that the higher the number of the disjoint paths, the greater the success rate. On the other hand, the assessment results show also that the higher the rate of attackers within the network, the less the success rate.

8 Conclusion

Currently, security threats in large scale autonomous P2P systems are increasingly present. Given that traditional security protocols fail to be applied in these systems free of central coordination points, we have proposed in this paper a new key exchange algorithm suitable to distributed systems.

It is not only about designing an interesting approach, it is also about a robust scheme. Indeed, the robustness goal is fulfilled by using multipath key exchange technique that extends both Diffie-Hellman protocol and Shamir's threshold in order to meet security expectations. In addition, based on disjoint paths and defined in order to route separately subkeys through the network, multipath routing methods are quite similar to Menger's theorem [3]. Finally, experiments show that our multipath key exchange scheme is robust to isolated MITM attacks and reduces substantially vulnerabilities to distributed MITM attacks as the number of disjoint paths increases.

References

[1] D. Ahmat, D. Magoni, and T. Bissyandé, "End-to-end key exchange through disjoint paths in P2P networks," in *European Alliance for Innovation, Endorsed Transaction on Security and Safety*, pp. 1–15, vol. 2-3, Jan. 2015.

[2] A. Avramidis, P. Kotzanikolaou, C. Douligeris, and M. Burmester, "Chord-pki: A distributed trust infrastructure based on p2p networks," *Computer Networks*, vol. 56, pp. 378–398, Jan. 2012.

[3] T. Böhme, F. Göring, and J. Harant, "Menger's theorem," *Journal of Graph Theory*, vol. 37, no. 1, pp. 35–36, 2001.

[4] C. Cassagnes, T. Tiendrebeogo, D. Bromberg, and D. Magoni, "Overlay addressing and routing system based on hyperbolic geometry," in *Proceedings of the 16th IEEE Symposium on Computers and Communications*, pp. 294–301, 2011.

[5] M. Castro, P. Druschel, A. Ganesh, A. Rowstron, and D. S. Wallach, "Secure routing for structured peer-to-peer overlay networks," *SIGOPS Operating Systems Review*, vol. 36, pp. 299–314, Dec. 2002.

[6] T. Y. Chang, M. S. Hwang, and W. P. Yang, "An improved multi-stage secret sharing scheme based on the factorization problem," *Information Technology and Control*, vol. 40, no. 3, pp. 246–251, 2011.

[7] T. Y. Chang, M. S. Hwang, and W. P. Yang, "An improvement on the lin-wu (t,n) threshold verifiable multi-secret sharing scheme," *Applied Mathematics and Computation*, vol. 163, no. 1, pp. 169–178, Apr. 2005.

[8] R. Chaubey and V. R. R. Manthena, *Decryption of Secure Sockets Layer Sessions Having Enabled Perfect Forward Secrecy Using a Diffie-Hellman Key Exchange*, Feb. 13, 2018. US Patent 9,893,883.

[9] S. F. Chiou, M. S. Hwang, and S. K. Chong, "A simple and secure key agreement protocol to integrate a key distribution procedure into the dss," *International Journal of Advancements in Computing Technology*, vol. 4, no. 19, pp. 529–535, Octo. 2012.

[10] H. Deng, A. Mukherjee, and D. P. Agrawal, "Threshold and identity-based key management and authentication for wireless ad hoc networks," in *Proceedings of the International Conference on Information Technology: Coding and Computing (ITCC'04)*, pp. 107, 2004.

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

[12] N. Döttling and S. Garg, "Identity-based encryption from the diffie-hellman assumption," in *Annual International Cryptology Conference*, pp. 537–569, 2017.

[13] Y. E. H. Shehadeh, O. Alfandi, and D. Hogrefe, "Towards robust key extraction from multipath wireless channels," *Journal of Communications and Networks*, vol. 14, no. 4, pp. 385–395, 2012.

[14] D. B. A. Epstein, "Curves on 2-manifolds and isotopies," in *Acta Math*, pp. 15–16, 1966.

[15] A. Escala, G. Herold, E. Kiltz, C. Rafols, and Jorge Villar, "An algebraic framework for diffie–hellman assumptions," *Journal of Cryptology*, vol. 30, no. 1, pp. 242–288, 2017.

[16] P. M. Fathimal and P. A. J. Rani, "Threshold secret sharing scheme for compartmented access structures," *International Journal of Information Security and Privacy*, vol. 10, no. 3, p. 9, 2016.

[17] C. E. Gero, J. N. Shapiro, and D. J. Burd, *Providing Forward Secrecy in a Terminating ssl/tls Connection Proxy Using Ephemeral Diffie-Hellman Key Exchange*, Dec. 27 2016. US Patent 9,531,685.

[18] C. Guo and C. C. Chan, "A novel threshold conference-key agreement protocol based on generalized chinese remainder theorem," *International Journal of Network Security*, vol. 17, no. 2, pp. 165–173, Mar. 2015.

[19] P. Harsha, P. Chanakya, and V. C. Venkaiah, "A reusable multipartite secret sharing scheme based on superincreasing sequence," *International Journal of Network Security*, vol. 20, no. 3, pp. 527–535, May 2018.

[20] B. King, "A dynamic threshold decryption scheme using bilinear pairings," *International Journal of Network Security*, vol. 17, no. 6, pp. 771–778, Nov. 2015.

[21] J. Kong, Z. Petros, H. Luo, S. Lu, and L. Zhang, "Providing robust and ubiquitous security support for mobile ad-hoc networks," in *Ninth International Conference on Network Protocols*, pp. 251–260, 2001.

[22] B. Koziel, A. Jalali, R. Azarderakhsh, D. Jao, and M. Mozaffari-Kermani, "Neon-sidh: Efficient implementation of supersingular isogeny diffie-hellman key exchange protocol on arm," in *International Conference on Cryptology and Network Security*, pp. 88–103, 2016.

[23] H. Kwon, S. Koh, J. Nah, and J. Jang, "The secure routing mechanism for dht-based overlay network," in *10th International Conference on Advanced Communication Technology (ICACT'08)*, vol. 2, pp. 1300–1303, 2008.

[24] J. Liu and J. Li, "A better improvement on the integrated diffie-hellman-dsa key agreement protocol," *International Journal of Network Security*, vol. 11, no. 2, pp. 114–117, Sep. 2010.

[25] D. Magoni and J. J. Pansiot, "Internet topology modeler based on map sampling," in *Proceedings of the 7th IEEE Symposium on Computers and Communications*, pp. 1021–1027, 2002.

[26] T. Mefenza and D. Vergnaud, "Polynomial interpolation of the generalized diffie–hellman and naor–reingold functions," *Designs, Codes and Cryptography*, pp. 1–11, 2018.

[27] H. T. Pan, J. R. Sun, and M. S. Hwang, "Cryptanalysis of biswas's multi-party keys scheme based on the diffie-hellman technique," *International Conference on Advances in Mechanical Engineering and Industrial Informatics*, Jan. 2015.

[28] H. K. Pathak and M. Sanghi, "Simple three party key exchange protocols via twin diffie-hellman problem," *International Journal of Network Security*, vol. 15, no. 4, pp. 256–264, July 2013.

[29] Q. Peng and Y. Tian, "A publicly verifiable secret sharing scheme based on multilinear diffie-hellman assumption," *International Journal of Network Security*, vol. 18, no. 6, pp. 1192–1200, Nov. 2016.

[30] C. Rajarama, J. N. Sugatoor, and T. Y. Swamy, "Diffie-hellman type key exchange, elgamal like encryption/decryption and proxy re-encryption using circulant matrices," *International Journal of Network Security*, vol. 20, 2018.

[31] J. Sen, "A multi-path certification protocol for mobile ad hoc networks," in *The 4th International Conference on Computers and Devices for Communication*, pp. 1–4, 2009.

[32] A. Shamir, "How to share a secret," *Communications of the ACM*, vol. 22, pp. 612–613, Nov. 1979.

[33] A. Shamir, "Identity-based cryptosystems and signature schemes," in *Proceedings of Crypto 84 on Advances in Cryptology*, pp. 47–53, 1984.

[34] M. Srivatsa and L. Liu, "Vulnerabilities and security threats in structured overlay networks: a quantitative analysis," in *The 20th Annual Computer Security Applications Conference*, pp. 252–261, 2004.

[35] Y. Takano, N. Isozaki, and Y. Shinoda, "Multipath key exchange on p2p networks," in *The First International Conference on Availability, Reliability and Security*, pp. 8, 2006.

[36] A. N. Tentu, V. K. Prasad, and V. C. Venkaiah, "Secret sharing schemes for multipartite access structures," *International Journal of Applied Engineering Research*, vol. 11, no. 7, pp. 5244–5249, 2016.

[37] T. Tiendrebeogo, D. Ahmat, D. Magoni, and O Sié, "Virtual connections in p2p overlays with dht-based name to address resolution," *International Journal on Advances in Internet Technology*, vol. 5, no. 1, pp. 11–25, 2012.

[38] M. S. Hwang, T. Y. Chang and W. P. Yang, "An improved multi-stage secret sharing scheme based on the factorization problem," *Information Technology and Control*, vol. 40, no. 3, pp. 246 – 251, 2010.

[39] R. Y. V. Subba and C. Bhagvati, "Crt based threshold multi secret sharing scheme," *International Journal of Network Security*, vol. 16, no. 4, pp. 249–255, July 2014.

[40] L. Valenta, D. Adrian, A. Sanso, S. Cohney, J. Fried, M. Hastings, J. A. Halderman, and N. Heninger, "Measuring small subgroup attacks against diffie-hellman (eprint)," in *Proceedings of 39th IEEE Symposium on Security and Privacy (Oakland'18)*, 2018. (`https://eprint.iacr.org/2016/995.pdf`)

[41] P. Wang, I. Osipkov, and Y. Kim, *Myrmic: Secure and Robust DHT Routing*, 2007. (`https://www.dtc.umn.edu/publications/reports/2006_20.pdf`)

[42] C. C. Yang, T. Y. Chang, and M. S. Hwang, "A (t,n) multi-secret sharing scheme," *Applied Mathematics and Computation*, vol. 151, no. 2, pp. 483–490, Apr. 2004.

Biography

Dr. Daouda Ahmat completed his Computer Science PhD studies in 2014 at the University of Bordeaux, France. He also graduated (MSc) from the same University in 2011 after having obtained a Bachelor degree from the University of NDjamena, Chad. He is currently the Vice-Rector of Virtual University of Chad. His research interests include Network Security, Mobile VPN, Key Exchange in Distributed Systems, Peer-to-Peer Networks,

Anonymous Systems, Blockchain and ICT for Development.

Mr. Marayi Choroma is an Industrial and Computer Engineer. He obtained a University Diploma of Research in Digital Education from the University of Lille 1, France in 2015. He is currently interested in the design and implementation of digital instruments for learning and teaching, including the integration of MOOCs into higher education programs in Chad.

Dr. Tégawendé F. Bissyandé is a research scientist at the Interdisciplinary Centre for Security, Reliability and Trust of the University of Luxembourg. He received his PhD degree in Computer Sciences from the University of Bordeaux in 2013. His main research interests are in software engineering, notably debugging software to repair bugs, patch vulnerabilities and identify security threats such as malware.

Efficient Access Control Scheme with Certificateless Signcryption for Wireless Body Area Networks

GaiMei Gao[1,2], XinGuang Peng[1], and LiZhong Jin[3]
(Corresponding author: XinGuang Peng)
College of Information and Computer, Taiyuan University of Technology[1]
No.79, West Yingze Street, Taiyuan, Shanxi, China
Department of Computer Science and Technology, Taiyuan University of Science and Technology[2]
Applied Science College, Taiyuan University of Science and Technology[3]
No.66, Waliu Road, Wanbailin District, Taiyuan, Shanxi, China
(Email: sxgrant@126.com)
(Received Jan. 9, 2018; Revised and Accepted May 5, 2018; First Online Jan. 13, 2019)

Abstract

Wireless body area networks (WBANs) can collect patients' vital data of body parameters and environment parameters via small wearable or implantable sensors. To ensure the security of the vital data, an efficient access control scheme with certificateless signcryption(CLSC) is designed. The correctness of the scheme is proved by mathematical calculation. It also proves that the scheme offers confidentiality and unforgeability in the random oracle model on the basis of the hardness of the Computational Diffie-Hellman (CDH) problem and Discrete Logarithm (DL) problem respectively. Compared with the existing three access control schemes utilizing signcryption, the scheme can satisfy more security properties and has the shortest computational time and the least energy consumption for the controller.

Keywords: Access Control; Certificateless; Signcryption;Wireless Body Area Networks

1 Introduction

Wireless body area networks (WBANs) can acquire human body's vital signals through a network which consists of intelligent and low-power micro- and nano-sensors and actuators. These sensors for collecting timely data can be placed on the body or implanted in the human body (or even in the blood stream). In addition to saving lives, WBANs is prevalent in reducing health care costs by removing the costly in-hospital monitoring of patients. In IEEE 802.15.6 [13], WBANs applications are classified into two types: medical and non-medical applications. In the study, we focus on the technological requirements of medical WBANs.

Security and privacy are two important considerations in WBANs. Since the patient-related data in the WBANs plays a critical role in medical diagnosis and treatment, it is necessary to ensure the security of these data in such a way that only authorized users can access these data [4, 18, 19]. Another aspect which should be considered in WBANs is the limitation of the controller's resources, especially storage space and computational capability. In order to protect the data privacy and reduce the energy consumption of computation and communication, lightweight access control schemes are needed. The certificateless public key cryptography (CL-PKC) [5] does not require the use of the certificate which brings the burden of certificate management, and CL-PKC avoids the key escrow problem because the user's private key is not generated by himself but by the user and the key generation center (KGC). Signcryption [3], as a cryptographic technique, can provide both the functions of public key encryption and digital signature in a logical single step at a significantly lower cost compared to traditional signature-then-encryption methods. A signcryption scheme can achieve confidentiality, authentication, integrity, and non-repudiation simultaneously at a lower cost. Therefore, we design an efficient access control with certificateless signcryption (CLSC) to protect data privacy of WBANs while reducing the computational overhead and storage overhead of resource-constrained controller. Many certificateless cryptosystems [1, 9, 10], such as certificateless encryption schemes, certificateless signcryption schemes, and certificateless access control schemes were proposed.

Access control is an important part of defense for the security of network systems, which protects data security and user privacy through only authorized users can access the WBANs. Some important progresses have been made in the access control for the WBANs. In

2011, Cagalaban and Kim [2] proposed a novel efficient access control scheme for the WBANs based on identity-based signcryption (IBSC) [12] (hereafter called CK). The signcryption method adopted in the CK scheme can simultaneously authenticate the users and protect the request messages. The scheme effectively solves the problem of a single point of failure in the traditional public-key infrastructure-supported system (PKI) by providing key generation and key management services without any assumption of pre-fixed trust relationship between network devices. However, CK has the key escrow problem since it is based on the IBSC. In 2016, Li and Hong [8] demonstrated an efficient certificateless access control scheme for the WBANs by using certificateless signcryption (CLSC) with public verifiability and ciphertext authenticity (hereafter called LH). The scheme can solve the key escrow problem and avoid the use of public key certificates. The controller could verify the validity of a ciphertext before decryption. Then Li *et al.* [7] proposed a novel certificateless signcryption scheme and designed a cost-effective and anonymous access control scheme for the WBANs with the novel signcryption (hereafter called LHJ). They reported that the proposed access control scheme achieved various securities and had the least computational cost and total energy consumption of the controller. However, the above two schemes may not be good choices since they require some costly bilinear pairing operations. The computational cost of a bilinear pairing operation is approximately twenty times higher than that of scale multiplication [6]. These costly operations are a heavy burden for resource-limited sensor nodes.

In this paper, we proposed an efficient access control scheme with certificateless signcryption for WBANs. The main contributions are:

1) A CLSC scheme without using bilinear pairing operation is proposed, and an efficient access control scheme for WBANs is constructed. The use of CL-PKC eliminates the burden of certificate management and solves the key escrow problem.

2) The correctness of the CLSC scheme is verified from the aspects of the partial key, the ciphertext and the signature.

3) It is formally proved that the scheme is semantically secure against indistinguishability-certificateless signcryption-adaptive chosen ciphertext attacks (IND-CLSC-CCA2) based on the hardness of the Computational Diffie-Hellman (CDH) problem and existential unforgeability-certificateless signcryption-chosen message attack (EUF-CLSC-CMA) based on the hardness of the Discrete Logarithm (DL) problem.

4) The security attributes of the scheme are analyzed.

5) Compared with three other access control schemes utilizing signcryption, the scheme is characterized by the lowest computational cost and energy consumption for the controller.

2 Preliminary

In this section, we present some mathematical assumptions, the security model and the network model.

2.1 Computational Assumptions

Definition 1. *Computational Diffie-Hellman (CDH). Given a 3-tuple* (p, aP, bP) *for two unknown elements* $a, b \in Z_q^*$, *here* G *is a group with prime order* q *and* P *is a generator of* G, *the CDH problem is to compute the value* abP *from* aP *and* bP. *The advantage of any probabilistic polynomial time algorithm* A *in solving the CDH problem in* G *is defined as* $Adv_A^{CDH} = Pr[A(p, aP, bP) = abP | a, b \in Z_q^*]$. *The CDH assumption is that the advantage* Adv_A^{CDH} *is negligibly small for any probabilistic polynomial time algorithm* A.

Definition 2. *Discrete Logarithm (DL). Given a 2-tuple* $(P, \mu P)$ *for an unknown element* $\mu \in Z_q^*$, *here* G *is a group with prime order* q *and* P *is a generator of* G, *the DL problem is to find the value* μ. *The advantage of any probabilistic polynomial time algorithm* A *in solving the DL problem in* Z_q^* *is defined as* $Adv_A^{DL} = Pr[A(P, \mu P) = \mu | \mu \in Z_q^*]$. *The DL assumption is that the advantage* Adv_A^{DL} *is negligibly small for any probabilistic polynomial time algorithm* A.

2.2 Security Model

All CLSC schemes may be subjected to two types of attacks [20]: Type-I adversary A_1 and Type-II adversary A_2.

Type-I adversary: The adversary A_1 is not accessible the master key, but he can replace public keys at his will. Therefore, the adversary A_1 is also called malicious user.

Type-II adversary: The adversary A_2 is accessible to the master key, but he cannot replace user's public keys. It represents a malicious KGC who generates partial private key of users.

Definition 3. *Confidentiality. A certificateless signcryption scheme is semantically secure against indistinguishability-certificateless signcryption-adaptive chosen ciphertext attacks (IND-CLSC-CCA2) if there is not a probabilistic polynomial time adversary* $A_{i(i=1,2)}$ *that has the non-negligible advantage in winning the game [20].*

Definition 4. *Unforgeability. A certificateless signcryption scheme is semantically secure against existential unforgeability-certificateless signcryption-chosen message attack (EUF-CLSC-CMA) if there is not a probabilistic polynomial time adversary* $A_{i(i=1,2)}$ *with the non-negligible advantage in winning the game [20].*

2.3 Network Model

The IEEE 802.15.6 working group has considered WBANs to operate in a one-hop or two-hop star topology. The node being placed on a location like the waist is the center of the star topology and controls the communication in WBANs [13]. Here we consider the one-hop star topology and all nodes in the WBANs are directly connected to the controller which all nodes talk. The WBANs contains some sensor nodes and a controller. Sensor nodes in, on or around the body collect vital signals of the patient and regularly transfer them to the corresponding controller. The controller aggregates information from the sensor nodes and communicates with the Internet. Figure 1 shows the overview of the network model of our WBANs applications. The framework is mainly composed of three entities: a Server Provider (SP), the WBANs of a patient, and a user (e.g. a physician, a researcher or an emergency). The SP deploys the WBANs and is responsible for the registration both of users and patients. The SP plays the role of KGC in the CLCS scheme and produces the partial key for any entity which registers at the SP. We suppose that the SP is honest. However, in practices, we do not need to fully trust the SP since it only knows the partial private key of the entity.

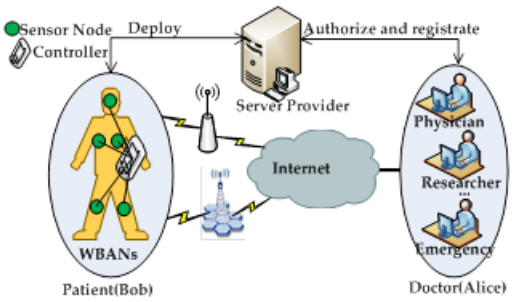


Figure 1: Network model of our WBANs applications

Here's a practical example. We assume that a patient Bob is hospitalized and the SP has deployed the WBANs of Bob. Bob's private key has generated when he registered at the SP. Sensor nodes in WBANs collect Bob's profile and medical records and transfer them to the controller. Doctor Alice has registered at the SP, and the SP has allocated expire data for Alice. When Alice needs to access the data of Bob, she first sends an access request message to Bob. Then Bob checks whether the Alice has the access privilege to his medical data. If Alice is authorized, Alice communicates with Bob to get the vital sign data in order to provide the better medical care service. Otherwise, Bob refuses the access request.

3 Construction of the Access Control Scheme

In this section, we first propose a CLSC scheme without using bilinear pairing operation. Then we construct an efficient access control scheme with the proposed CLSC scheme.

3.1 The Proposed CLSC Scheme

The CLSC scheme Π = (Setup, PartialKeyGen, KeyGen, Sign, UnSign) consists of five algorithms.

Setup: Given a security parameter k, the SP chooses cyclic group G of a large prime order q, a generator P of G, and three security hash functions $H_1 : \{0,1\}^* \times G \rightarrow Z_q^*$, $H_2 : \{0,1\}^* \rightarrow Z_q^*$ and $H_3 : Z_q^* \rightarrow \{0,1\}^{l_0+|Z_q^*|}$. Here l_0 is the number of bits of a message to be sent, and $|Z_q^*|$ is the number of bits of the element in Z_q^*. Then the SP selects the system's master key $z \in Z_q^*$ at random and computes the corresponding public key $y = zP$. Finally, the SP distributes the system parameters $params = (G, q, P, y, H_1, H_2, H_3)$ and keeps the master key z secretly.

Partial Key Generation (PartialKeyGen): When entities want to register his/her identity ID_i to the SP, he/she first sends ID_i to the SP. Then the SP selects random number $r_i \in Z_q^*$, computes $R_i = r_iP$ and $d_i = r_i + zH_1(ID_i, R_i)$. Finally, the SP sets d_i as the entity's partial private key and R_i as the entity's partial public key, and transfers (d_i, R_i) to the entity over a confidential and authentic channel.

Key Generation (KeyGen): When the entity receiving the partial key generated by SP, he/she needs to choose another part of key and generate his/her full key. The entity selects secret value $x_i \in Z_q^*$ at random and computes $X_i = x_iP$. Then the entity sets $SK_i = (d_i, x_i)$ as his/her private key and $PK_i = (R_i, X_i)$ as his/her public key.

Here, we assume that the access request is sent by doctor Alice whose identity is ID_A, and the receiver is patient Bob whose identity is ID_B in our CLSC scheme. Alice's public key is $PK_A = (R_A, X_A)$ and private key is $SK_A = (d_A, x_A)$. Bob's public key is $PK_B = (R_B, X_B)$ and private key is $SK_B = (d_B, x_B)$. Alice and Bob can verify the correctness of the partial private key and partial public key with the equation $r_AP + H_1(ID_A, R_A)y = d_AP$ and $r_BP + H_1(ID_B, R_B)y = d_BP$ respectively.

Signcryption (Sign): With the system parameters , access plaintext message m, Alice's identity ID_A and private key $SK_A = (d_A, x_A)$, Bob's identity ID_B and public key $PK_B = (R_B, X_B)$, Alice runs following steps to generate the ciphertext $\delta = (s, C, T)$.

1) Selects a random $\beta \in Z_q^*$ and computes $T = \beta P$.
2) Computes $h_1 = H_1(ID_B, R_B)$.
3) Computes $V_A = \beta(X_B + R_B + h_1 y)$.
4) Computes $h = H_2(m||T||ID_A||ID_B||X_A||X_B)$.
5) Computes $s = (x_A + \beta)/(h + d_A + x_A)$.
6) Computes $C = H_3(V_A) \oplus (m||s)$.
7) Outputs a ciphertext $\delta = (s, C, T)$.

UnSigncryption (UnSign): Taking a ciphertext δ, Bob's identity ID_B and private key $SK_B = (d_B, x_B)$, Alice's identity ID_A and public key $PK_A = (R_A, X_A)$ as inputs, Bob execute following steps to complete the verification of signcryption.

1) Computes $V_B = (x_B + d_B)T$.
2) Recover the message $m||s = H_3(V_B) \oplus C$, and complete decryption.
3) Computes $h = H_2(m||T||ID_A||ID_B||X_A||X_B)$.
4) Computes $h_1' = H_1(ID_A, R_A)$.
5) If $s(X_A + R_A + h_1' \cdot y + h \cdot P) = X_A + T$ holds, the message m is valid and Alice communicates with Bob using the session key $H_3(V_A)$ or $H_3(V_B)$. Otherwise return $\perp$.

3.2 Our Access Control Scheme

In this section, with the proposed CLSC scheme, we design an efficient access control scheme with certificateless signcryption for the WBANs. The scheme has four phases: the initialization phase, the registration phase, the authentication and authorization phase, and the revocation phase. We define ED as an expiration date. The access control scheme is summarized in Figure 2.

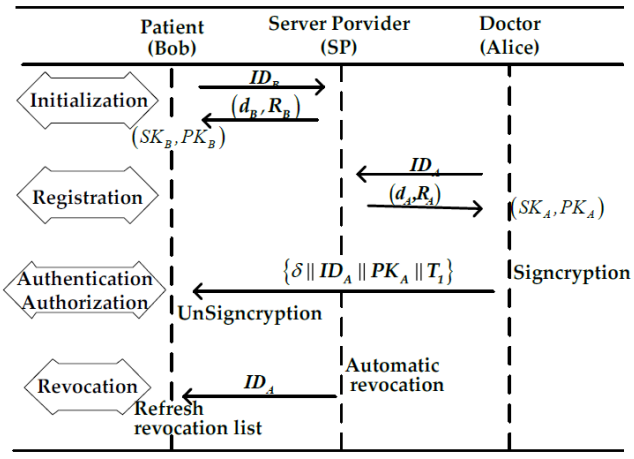


Figure 2: Certificateless access control scheme

3.2.1 Initialization Phase

In this phase, the SP runs Setup algorithm to deploy the WBANs and generate the system parameters. The patient Bob with identity ID_B gets his/her public key $PK_B = (X_B, R_B)$ and private key $SK_B = (x_B, d_B)$. In particular, Bob's communications with Internet are all done by the controller of the WBANs, so Bob also refers to the controller of the WBANs. The SP may run Setup algorithm and PartialKeyGen algorithm.

3.2.2 Registration Phase

Only when the doctor Alice is a registered user of the SP can she access the data of patient Bob. Alice submits his identity ID_A to the SP and then the SP checks whether the identity is valid. If not, the SP rejects the registration request. Otherwise, the SP sets an expiration date ED and runs PartialKeyGen algorithm to produce a partial private key (d_A, R_A). After receiving (d_A, R_A), Alice runs KeyGen algorithm to get the full private key $SK_A = (d_A, x_A)$ and the full public key $PK_A = (R_A, X_A)$.

3.2.3 Authentication and Authorization Phase

When the doctor Alice with the identity ID_A wants to access the monitoring data of the WBANs, Alice firstly produces a request message m and runs Sign algorithm to generate a ciphertext $\delta = (s, C, T)$. To resist the replay attack, we may concatenate the request message and a timestamp to form a new signcrypted message. Then Alice sends the requirement message $\{\delta||ID_A||PK_A||T_1\}$ to Bob, wherein T_1 is the current timestamp. When obtaining the access request from Alice, Bob checks $T_2 - T_1 < \Delta T$ whether holds, wherein T_2 is the current timestamp. If it does not hold, Bob terminates the session. Otherwise, Bob runs Unsign algorithm to complete unsigncryption. When the return value of Unsign algorithm is $\perp$, Bob rejects the request. Otherwise, the request is valid and Alice communicates with Bob using the session key $H_3(V_A)$ or $H_3(V_B)$. This session key has been established between Bob and Alice.

3.2.4 Revocation

The access privilege is automatically revoked by the expired date ED. For example, if the expired date ED is "2017-12-31", the user only can access the WBANs before December 31, 2017. That is to say, the SP will revoke Alice's partial private key and partial public key, which made Alice automatically illegal after December 31, 2017. For some reasons we need to revoke the Alice's access privilege before the expired date, the SP will submit the Alice's identity to Bob, which keeps a list of revoked identities for identifying the validity of users. Bob will add a record to his revocation list and this makes Alice an illegal user.

4 Performance Analysis

In this section, we will analysis the access control scheme. First is the validation of mathematical correctness. Then we demonstrate the scheme is provably secure based on CDH problem and DL problem. Third is the analysis of security property. Finally is the efficiency comparison with three other schemes.

4.1 Correctness of the Proposed CLSC Scheme

4.1.1 Correctness of the Partial Key

Both of Alice and Bob can verify the correctness of the partial key (d_i, R_i) which SP assigned to him/her by following equal.

$$\begin{aligned} & R_i + H_1(ID_i, R_i)y \\ = \; & r_iP + zPH_1(ID_i, R_i) \\ = \; & (r_i + zH_1(ID_i, R_i))P \\ = \; & d_iP. \end{aligned}$$

4.1.2 Correctness of the Ciphertext

The verification of the ciphertext in the UnSign algorithm is obtained from the following:

$$\begin{aligned} V_B &= (X_B + d_B)T \\ &= (x_B + r_B + zH_1(ID_B, R_B))\beta P \\ &= \beta(X_B + R_B + yH_1(ID_B, R_B)) \\ &= V_A. \end{aligned}$$

We will then obtain the following:

$$\begin{aligned} m||s &= H_3(V_B) \oplus C \\ &= H_3(V_A) \oplus H_3(V_A) \oplus (m||s) \\ &= m||s. \end{aligned}$$

4.1.3 Correctness of the Signature

The verification operation of the signature in the UnSign algorithm can be completed by following equation.

$$\begin{aligned} & s(X_A + R_A + h_1'y + hP) \\ = \; & \frac{x_A + \beta}{h + d_A + x_A}(x_AP + r_AP + zH_1(ID_A, R_A)P + hP) \\ = \; & \frac{x_A + \beta}{h + d_A + x_A}(x_A + d_A + h)P \\ = \; & (x_A + \beta)P \\ = \; & x_AP + \beta P \\ = \; & X_A + T. \end{aligned}$$

4.2 Proof of Security

Based on the CDH problem and DL problem in the random oracle model, we prove that the CLSC scheme satisfies confidentiality in the following Theorem 1 and Theorem 2, and unforgeability in the following Theorem 3.

Theorem 1. *(Type-I Confidentiality): In the random oracle model, if there is an adversary A_1 who can win the IND-CLSC-CCA2 game with non-negligible advantage ε, there will be an algorithm $\mathbb{F}$ which can solve the CDH problem with an advantage $Adv_{A_1}^{IND-CLSC-CCA2} \geq \frac{\varepsilon}{q_1^2 q_3}(1 - \frac{1}{q_s+1})^{q_s}\frac{1}{q_s+1}$. Here, the adversary A_1 performs at most q_i hash queries to random oracles $H_{i(i=1,2,3)}$ and q_s signcryption queries.*

Proof. Supposing that there is an adversary A_1 who can break our CLCS scheme. We want to build an algorithm $\mathbb{F}$ that use A_1 to solve CDH problem. The algorithm $\mathbb{F}$ receives an instance (P, aP, bP) of CDH problem to compute abP. $\mathbb{F}$ respectively maintains the lists $L_1, L_2, L_3, L_D, L_{SK}, L_{PK}, L_S, L_U$ to track the oracle model H_1, H_2, H_3, partial key generation, private key generation, public key generation, signcryption, and unsigncryption. Moreover, $\mathbb{F}$ sets the list L_{rec} to record the parameters of the challenge identity. Each list is empty at the beginning.

Setup: Input security parameter k. $\mathbb{F}$ executes Setup algorithm and sends the generated parameters $params = (G, q, P, y, H_1, H_2, H_3)$ to A_1. $\mathbb{F}$ can also simulate the partial key generation, key generation, public key query, public key replacement, signcryption, and un-signcryption oracle to provide responses to A_1's queries.

Find Stage: A_1 can adaptively make a polynomial bounded number of the following queries.

1) H_1 queries: When $\mathbb{F}$ receives the query $H_1(ID, R)$ from A_1, if (ID, R, h_1, c) exists in the list L_1, $\mathbb{F}$ returns h_1 to A_1. Otherwise, $\mathbb{F}$ selects random $c \in \{0, 1\}$, here $Pr[c = 1] = \delta = 1/(q_s + 1)$ [21]. When $c = 0$, $\mathbb{F}$ randomly chooses $h_1 \in Z_q^*$, returns it to A_1, and inserts (ID, R, h_1, c) into the list L_1. When $c = 1$, $\mathbb{F}$ lets $h_1 = k$ and returns k to A_1.
2) H_2 queries: When $\mathbb{F}$ receives the query $H_2(m, T, ID_A, ID_B, X_A, X_B)$ from A_1, if $(m, T, ID_A, ID_B, X_A, X_B, h_2)$ exists in the list L_2, $\mathbb{F}$ returns h_2 to A_1. Otherwise, $\mathbb{F}$ randomly selects $h_2 \in Z_q^*$, and returns it to A_1. Then $\mathbb{F}$ inserts $(m, T, ID_A, ID_B, X_A, X_B, h_2)$ into the list L_2.
3) H_3 queries: When $\mathbb{F}$ receives the query $H_3(T)$ from A_1, if (T, h_3) exists in the list L_3, $\mathbb{F}$ returns h_3 to A_1. Otherwise, $\mathbb{F}$ randomly picks $h_3 \in Z_q^*$, and returns it to A_1. Then $\mathbb{F}$ inserts (T, h_3) to the list L_3.
4) Partial Key queries: A_1 submits a request (ID, d, R). $\mathbb{F}$ checks whether the (ID, d, R) already exists in the list L_D. If it exists, $\mathbb{F}$ returns (d, R) to A_1, otherwise, since $\mathbb{F}$ does not know the master secret key, $\mathbb{F}$ randomly selects $r, z \in Z_q^*$, and computes the entity's partial

private key as $d = r + zH_1(ID, R)$. $\mathbb{F}$ inserts (ID, d, R) into the list L_D and returns (d, R) to A_1.

5) Private Key queries: A_1 submits a request (ID, d, x). $\mathbb{F}$ checks whether the (ID, d, x) already exists in the list L_{SK}. If it exists, $\mathbb{F}$ returns (d, x) to A_1. Otherwise, $\mathbb{F}$ obtains d by partial key queries, then randomly picks $x \in Z_q^*$, inserts (ID, d, x) into the list L_{SK}, and finally returns (d, x) to A_1.

6) Public Key queries: A_1 submits a request (ID, R, X). $\mathbb{F}$ responds as follows:
   - If (ID, R, X) already exists in the list L_{PK}, $\mathbb{F}$ returns (R, X) to A_1.
   - Otherwise, $\mathbb{F}$ checks the list L_d and L_{SK}. If there is the record of the entity with ID, $\mathbb{F}$ can obtain (R, x), then compute $X = xP$, insert (ID, R, X) into L_{PK}, and returns (R, X) to A_1 as a response. If there is no record of ID in the list L_d and L_{SK}, $\mathbb{F}$ checks the list L_1. If $c = 1$, $\mathbb{F}$ randomly picks $r, x \in Z_q^*$, computes $R = rP$, $X = xP$, inserts (ID, R, X) into L_{PK}, and returns (R, X) to A_1. At the same time, $\mathbb{F}$ inserts (ID, r, x, c) into L_{rec}. If $c = 0$, $\mathbb{F}$ runs private key queries, obtains (R, X), inserts (ID, R, X) into L_{PK}, and returns (R, X) to A_1.

7) Replace Public Key queries: A_1 supplies identity ID and a new public key (R', X'). $\mathbb{F}$ replaces the current public key (R, X) by the new key (R', X').

8) Signcryption queries: A_1 supplies two identities (ID_A, ID_B) and a message m. $\mathbb{F}$ checks the (ID_A, R_A) in the list L_1 and responds as follows:
   a. If $c = 0$, $\mathbb{F}$ gets (ID_A, d_A, x_A), (ID_B, R_B, X_B) respectively from the list L_{SK}, L_{PK} according to ID_A, ID_B, runs the Sign algorithm to complete signcryption, and returns ciphertext $\delta = (s, C, T)$ to A_1.
   b. If $c = 1$, $\mathbb{F}$ fails and aborts.

9) Un-Signcryption queries: A_1 supplies two identities (ID_A, ID_B) and a ciphertext $\delta = (s, C, T)$. $\mathbb{F}$ checks the (ID_B, R_B) in the list L_1 and responds as follows:
   a. If $c = 0$, $\mathbb{F}$ gets (ID_A, R_A, X_A), (ID_B, d_B, x_B) respectively from the list L_{PK}, L_{SK} according to ID_A, ID_B, runs the UnSign algorithm to complete unsigncryption, and returns the message m to A_1.
   b. If $c = 1$, $\mathbb{F}$ traverses down (V_B, h_3) of the list L_3, then computes $m||s = H_3(V_B) \oplus C$ and completes the un-signcryption. $\mathbb{F}$ selects h_1' from (ID_A, R_A, h_1', c) in the list L_1, selects R_A, X_A from (ID_A, R_A, X_A) in the list L_{PK}, selects h_2 from $(T, ID_A, ID_B, X_A, X_B, m, h_2)$ in the list L_2, where $h = h_2$, then $\mathbb{F}$ verifies whether the equation $s(X_A + R_A + h_1'y + hP) = X_A + T$ is valid. If the equation holds, then $\mathbb{F}$ outputs m, otherwise $\mathbb{F}$ starts from the next record of the list L_3 and redo Step b. If all the items in the list L_3 have not been returned, then $\mathbb{F}$ outputs $\perp$, which means un-signcryption fails.

Challenge Stage: A_1 can adaptively make two different messages m_0, m_1 with the same length and two challenge identities ID_A, ID_B. $\mathbb{F}$ firstly checks (ID_B, R_B) in the list L_1. If $c = 0$, $\mathbb{F}$ stops. Otherwise, $\mathbb{F}$ makes a Public Key queries to ensure that (x_B, r_B) already exist in the list L_{rec}. Then the algorithm $\mathbb{F}$ selects $s^*, c^* \in Z_q^*$ at random and sets $T^* = \beta P$. $\mathbb{F}$ sends the challenge ciphertext $\delta^* = (s^*, c^*, T^*)$ to A_1.

Guess Stage: A_1 can make a polynomial bounded number of queries like that in the Find stage. Finally, $\mathbb{F}$ outputs her guess c'. If $c' = c$, A_1 can make a query in H_3 with $V' = \beta(X_B + R_B + h_1 y)$. In this case, the candidate answer of the CDH problem is stored in the list L_3. $\mathbb{F}$ ignores the guessed value of A_1, selects V' from the list L_3 at random, and outputs $(V' - (x_B + r_B)T^*)/k = z\beta P$ as the answer to CDH problem, where x_B, r_B, T^*, V' are known to the algorithm $\mathbb{F}$. Otherwise, the algorithm $\mathbb{F}$ does not solve the CDH problem.

The algorithm $\mathbb{F}$ simulates the real attack situation for A_1. If $\mathbb{F}$ is not terminated in the process of simulation and can breach the confidentiality in this paper with non-negligible probability ε, $\mathbb{F}$ outputs the valid answer of the CDH problem.

Now, we evaluate the probability of success. The probability that A_1 runs partial private key queries or private key queries for ID_B is at least $1/q_1^2$. The probability that $\mathbb{F}$ successfully selects V' from the list L_3 as a candidate answer for the CDH problem is $1/q_3$. The non-termination probability is $(1-\delta)^{q_s}$ in the find stage. The non-termination probability is δ in the challenge stage. Therefore, the probability that $\mathbb{F}$ does not abort during the simulation is at least $\frac{\varepsilon}{q_1^2 q_3}(1 - \frac{1}{q_s+1})^{q_s}\frac{1}{q_s+1}$.

To sum up, if the algorithm $\mathbb{F}$ does not abort in the simulation process and A_1 can break the confidentiality of our signcryption scheme with the non-negligible advantage ε, $\mathbb{F}$ can output the valid solution of CDH problem with the advantage $Adv_{A_1}^{IND-CLSC-CCA2} \geq \frac{\varepsilon}{q_1^2 q_3}(1 - \frac{1}{q_s+1})^{q_s}\frac{1}{q_s+1}$. □

Theorem 2. *(Type-II Confidentiality): In the random oracle model, if there is an adversary A_2 who can win the IND-CLSC-CCA2 game with a non-negligible advantage*

ε, there will be an algorithm $\mathbb{F}$ that solves the CDH problem with an advantage $Adv_{A_2}^{IND-CLSC-CCA2} \geq \frac{\varepsilon}{q_1^2 q_3}(1-\frac{1}{q_s+1})^{q_s}\frac{1}{q_s+1}$. Here, the adversary A_2 performs at most q_i hash queries to random oracles $H_{i(i=1,2,3)}$ and q_s signcryption queries.

Proof. The proof idea is similar to Theorem 1 except the following aspects.

1) The adversary A_2 knows the system master key z.
2) In the Public Key queries, we set $R = zP$ other than $R = rP$, and we insert $(ID, -, x, c)$ into L_{rec} other than (ID, r, x, c).
3) In the guess stage, $\mathbb{F}$ outputs $V' - (x_B + kz)T^* = z\beta P$ as the answer to CDH problem.

□

Theorem 3. *(Unforgeability): In the random oracle model, if there is an adversary $A_{i(i=1,2)}$ who can win the EUF-CLSC-CMA game with non-negligible advantage ε, there will be an algorithm $\mathbb{F}$ that solves the DL problem with an advantage $Adv_{A_i(i=1,2)}^{EUF-CLSC-CMA} \geq \frac{\varepsilon}{9q_1^2}(1-\frac{1}{q_s+1})^{q_s}$. Here, the adversary $A_{i(i=1,2)}$ performs at most q_1 hash queries to random oracles $H_{i(i=1,2,3)}$ and q_s signcryption queries.*

Proof. Supposing that there is an adversary $A_{i(i=1,2)}$ who can break our CLCS scheme. We want to build an algorithm $\mathbb{F}$ which uses $A_{i(i=1,2)}$ to solve DL problem. The algorithm $\mathbb{F}$ receives an instance $(P, \mu P)$ of the DL problem and his goal is to compute μ.

Setup: The algorithm $\mathbb{F}$ set $y = \mu p$ for the adversary A_1. The other settings are the same as those in Theorem 1 for A_1. The algorithm $\mathbb{F}$ set $y = zp$ for the adversary A_2. The other settings are the same as those in Theorem 2 for A_2.

Queries: The adversary A_1 can adaptively make a polynomial bounded number of queries like those in Theorem 1, whereas the adversary A_2 can adaptively make the queries like those in Theorem 2.

Forgery: After a polynomial bounded number of queries, $A_{i(i=1,2)}$ outputs a faked ciphertext $\delta^* = (s^*, c^*, T^*)$ on message m^* with ID_A as the sender and ID_B as the receiver.

The algorithm $\mathbb{F}$ first checks the list L_1. If $c = 0$, $\mathbb{F}$ aborts. Otherwise, $\mathbb{F}$ can get the private key of ID_B, compute $V_B^* = (x_B + d_B)T^*$ and get h_3^* by H_3 queries with V_B^*. $\mathbb{F}$ recovers m^*, s^* by h_3^* and verifies the δ^*. If the $A_{i(i=1,2)}$ has successfully forged a user, $\mathbb{F}$ can get two legal signatures $(m^*, ID_A, ID_B, T^*, h, s_1)$ and $(m^*, ID_A, ID_B, T^*, h', s_1)$ with the Splitting Lemma [15], where $h \neq h'$. Thus, we get $T^* = \beta P = (s_1(h + d_A + x_A) - x_A)P = (s_2(h' + d_A + x_A) - x_A)P$ and $s_1(h + d_A + x_A)) = s_2(h' + d_A + x_A))$.

For Type-I attack A_1, it is $s_1(h + r_A + \mu k + x_A) = s_2(h' + r_A + \mu k + x_A)$, where $k = h_1 = H_1(ID_A, R_A)$. Only μ is unknown in this formula, so μ can be computed.

For Type-II attack A_1, it is $s_1(h + r_A + zk + x_A) = s_2(h' + r_A + zk + x_A)$, where $k = h_1 = H_1(ID_A, R_A)$. Only r_A is unknown in this formula, so r_A can be solved. We have set $R = r_A P = \mu P$ in the Public Key queries, so μ can be computed.

Now, we evaluate the probability of success. The probability that A_1 runs partial private key queries or private key queries for ID_A is at least $1/q_1^2$. The non-termination probability is $(1-\delta)^{q_s}$ in the find stage. The probability of failure is less than $1/9$ when two or more effective ciphertexts are produced with the oracle replay technique [15]. Therefore, the probability that $\mathbb{F}$ can solve the DL problem is at least $\frac{1}{9q_1^2}$. Thus, the probability that $\mathbb{F}$ successfully forges a user is at least $\frac{\varepsilon}{9q_1^2}(1-\frac{1}{q_s+1})^{q_s}$. □

4.3 Analysis of Security Properties

In the authentication and authorization phase, the session key is only known by the patient Bob and the doctor Alice, the scheme can achieve the confidentiality for future communication between them. In addition, the scheme uses the proposed CLSC scheme that is proved to have confidentiality in theorem1 and theorem2 and unforgeability in theorem 3, so the access control achieves confidentiality property and unforgeability property. The non-repudiation of the access request is guaranteed by introducing the timestamp. Owing to the characteristics of the CL-PKC, the access control can solve key escrow problem and avoid the use of public key certificates. When we design the CLSC scheme, we don't use bilinear pair operations, so our scheme avoids the bilinear pairing operation. Table 1 is the security properties comparing of the four schemes.

4.4 Efficiency Comparisons

In this section, we analyze the performance of our access control scheme in regard to energy consumption and communication overhead. Firstly, we compare the scheme with other three schemes of CK [2], LH [8] and LHJ [7] in computation efficiency and communication efficiency. The computation efficiency is determined by the computational cost of algorithm and the communication efficiency is determined by the length of ciphertext and public key. The symbol P denotes pairing operation, the symbol E denotes an exponentiation operation, the symbol M denotes a point multiplication operation. Let $|*|$ denote the length of element $*$. For example, $|G|$ denotes the length of element in group G and $|m|$ denotes the length of message space. As can be seen from Table 2, our scheme has the lower computational cost than the other three schemes for both Alice and Bob. Here, we neglect the cost of other operations because they are much smaller than the above three operations.

Table 1: Comparisons of security properties

	CK [2]	LH [8]	LHJ [7]	Our scheme
Confidentiality	√	√	√	√
Unforgeability	√	√	√	√
Authentication	√	√	√	√
Non-repudiation	√	√	√	√
No certificate	√	√	√	√
No key escrow	×	√	√	√
Without bilinear pairing	×	×	×	√

Abbreviations: √: Scheme prevents this attack or satisfies the attribute, ×: Scheme fails to prevent the attack or does not satisfy the attribute.

Table 2: Performance evaluation of the four schemes

Schemes	Computational Cost (Alice)	Computational Cost (Bob)	Communication Cost (Bob)
CK [2]	1P+3M	3P+M	$2\|G_1\| + \|ID\| + \|m\|$
LH [8]	2E	1P+1M+1E	$\|G_1\| + \|G_2\| + 3\|Z_p^*\| + \|ID\| + \|m\|$
LHJ [7]	1E+4M	2P+2M+1E	$3\|G_1\| + \|ID\| + \|m\|$
Ours	3M	4M	$5\|Z_q^*\| + \|ID\| + \|m\|$

Quantitative evaluation results for the four schemes are described below. Here, we only consider Bob's overhead, because his controller's resource is limited. We adopt the result in [14, 17] on the MICA2 mote which is equipped with an ATmega128 8-bit processor locked at 7.3728 MHz, 128KB ROM, and 4KB RAM. A pairing operation costs 1.9 s and an exponentiation operation costs 0.9 s by using a supersingular curve $y^2 + y = x^3 + x$ with an embedding degree of 4 and implementing an η_T pairing: $E(F_{2^{271}}) \times E(F_{2^{271}}) \rightarrow F_{2^{4271}}$, which is also equivalent to the 80-bit security level. According to the previous results [8], a point multiplication over the supersingular curve costs 0.81 s. Therefore, the computational time on the controller of CK [2], LH [8], LHJ [7], and our scheme are respectively $3 \times 1.9 + 1 \times 0.81 = 6.51$ s, $1 \times 1.9 + 1 \times 0.81 + 1 \times 0.9 = 3.61$ s, $2 \times 1.9 + 2 \times 0.81 + 1 \times 0.9 = 6.32$ s and $4 \times 0.81 = 3.24$ s. We also suppose that the power level of MICA2 is 3.0 V. The current draw in active mode is 8.0 mA and the current draw in receiving mode is 10 mA [15]. For energy consumption, according to the evaluation method [11,16], a pairing operation consumes $3.0 \times 8.0 \times 1.9 = 45.6$ mJ, a point multiplication operation consumes $3.0 \times 8.0 \times 0.81 = 19.44$ mJ, an exponentiation operation in G_2 consumes $3.0 \times 8.0 \times 0.9 = 21.6$ mJ. Therefore, the computational energy consumption on the controller of CK [2], LH [8], LHJ [7] and our scheme are $3 \times 45.6 + 1 \times 19.44 = 156.24$ mJ, $1 \times 45.6 + 1 \times 19.44 + 1 \times 21.6 = 86.64$ mJ, $2 \times 45.6 + 2 \times 19.44 + 1 \times 21.6 = 151.68$ mJ and $4 \times 19.44 = 77.76$ mJ respectively.

Figure 3 and Figure 4 respectively describe the computational time and energy consumption of the controller. It is clear that our scheme has the shortest computational time and least energy consumption among the four schemes.

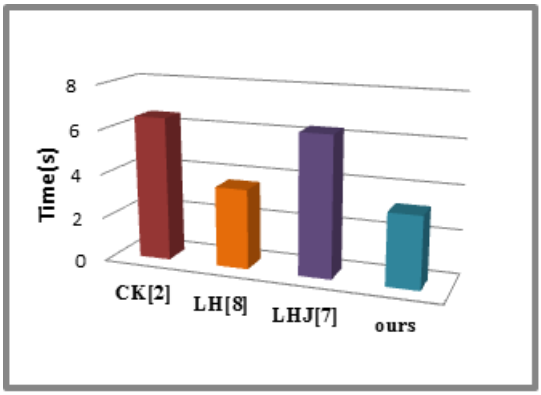


Figure 3: The computational time of the controller

For the communication cost, we suppose that $|m| = 160$ bits and $|ID| = 80$ bits. CK [2], LH [8], LHJ [7] schemes use a curve over the binary field $F_{2^{271}}$ with the G_1 of 252-bit prime order. As in [12, 17], the size of an element in group G_2 is 542 bits and can be compressed to 34 bytes. The size of an element in group G_2 is 1084 bits and can be compressed to 136 bytes. The size of an element of Z_q^* is 32 bytes. In CK [2], LH [8], LHJ [7] and our scheme, the controller needs to receive $2|G_1| + |ID| + |m|$ bits$=2 \times 34 + 10 + 20 = 98$ bytes, $|G_1| + |G_2| + 3|Z_p^*| + |ID| + |m|$ bits$=34 + 136 + 3 \times 32 + 10 + 20 = 296$ bytes, $3|G_1| + |ID| + |m|$ bits$=3 \times 34 + 10 + 20 = 132$ bytes, and $5|Z_q^*| + |ID| + |m|$ bits$=5 \times 32 + 10 + 20 = 190$ bytes respectively. From [12, 17], we know the controller

Table 3: Energy consumption of the four schemes

Schemes	Computational energy consumption (mJ)	Communication energy consumption (mJ)	Total energy consumption (mJ)
CK [2]	156.24	1.86	158.1
LH [8]	86.64	5.62	92.26
LHJ [7]	151.68	2.51	154.19
Ours	77.76	3.61	81.37

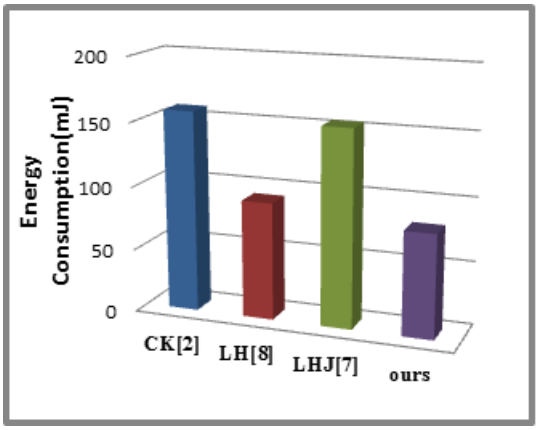


Figure 4: The energy consumption of the controller

takes $3 \times 10 \times 8/12400 = 0.019$ mJ to receive one-byte message. Therefore, in CK [2], LH [8], LHJ [7] and our scheme, communication energy consumption values of the controller are $0.019 \times 98 = 1.86$ mJ, $0.019 \times 296 = 5.62$ mJ, $0.019 \times 132 = 2.51$ mJ, and $0.019 \times 190 = 3.61$ mJ respectively. The total energy consumption of CK [2], LH [8], LHJ [7] and our schemes are $156.24 + 1.86 = 158.1$ mJ, $86.64 + 5.62 = 92.26$ mJ, $151.68 + 2.51 = 154.19$ mJ, and $77.76 + 3.61 = 81.37$ mJ respectively. Table 3 provides energy consumption of four schemes. Although the communication cost of our scheme is more than that of CK [2] and LHJ [7], the total energy consumption of our scheme is less than that of other three schemes. The controller's energy consumption of computation and communication in our scheme is almost half of that in CK [2] and LHJ [7].

5 Conclusions

In this paper, we proposed a new CLSC scheme without using bilinear pairing operation and constructed an efficient access control scheme using the proposed CLSC scheme for the WBANs. We verified the mathematical correctness of the CLSC scheme from the aspect of the partial key, the ciphertext and the signature. Then we proved that the proposed scheme offered confidentiality and unforgeability in the random oracle model on the basis of the hardness of the CDH problem and the DL problem respectively. Moreover, we have analyzed the security property and concluded that our scheme satisfy more security property than three others schemes. As far as performance analysis is concerned, our access control scheme had the shortest computational time and the least energy consumption compared with the existing three access control schemes utilizing signcryption.

Acknowledgements

This work is supported by the Key Research and Development Program of Shanxi Province under Grant No.201703D121042-1.

References

[1] S. K. Balakrishnan and V. P. Jagathy Raj, "Practical implementation of a secure email system using certificateless cryptography and domain name system", *International Journal of Network Security*, vol. 18, no. 1, pp. 99-107, 2016.

[2] G. Cagalaban and S. Kim, "Towards a secure patient information access control in ubiquitous healthcare systems using identity-based signcryption", in *Proceedings of 13th International Conference on Adv. Commun. Technol. (ICACT'11)*, pp. 863-867, 2011.

[3] L. Cheng and Q. Wen, "An improved certificateless signcryption in the standard model", *International Journal of Network Security*, vol. 17, no. 5, pp. 597-606, 2015.

[4] G.Gao, X. Peng, Y. Tian and Z. Qin, "A chaotic maps-based authentication scheme for wireless body area networks", *International Journal of Distributed Sensor Networks*, vol. 12, no. 7, pp. 2174720-2174720, 2016.

[5] M. Hassouna, B. Barry, and E. Bashier, "A new level 3 trust hierarchal certificateless public key cryptography scheme in the random oracle model", *International Journal of Network Security*, vol. 19, no. 4, pp. 551-558, 2017.

[6] D. He, J. Chen, and J. Hu, "An ID-based proxy signature schemes without bilinear pairings", *Annals of Telecommunications*, vol. 66, no. 11-12, pp. 657-662, 2011.

[7] F. Li, Y. Han, and C. Jin, "Cost-effective and anonymous access control for wireless body area networks",

IEEE Systems Journal, vol.12, no. 1, pp. 747-758, 2018.

[8] F. Li and J. Hong, "Efficient certificateless access control for wireless body area networks", *IEEE Sensors Journal*, vol. 16, no. 13, pp. 5389-5396, 2016.

[9] F. Li, J. Hong, and A. A. Omala, "Efficient certificateless access control for industrial Internet of things", *Future Generation Computer Systems*, vol. 76, pp.285-292, 2017.

[10] M. Luo, Y. Wan, and D. Huang, "Certificateless hybrid signcryption scheme with known session-specific temporary information security", *International Journal of Network Security*, vol. 19, no. 6, pp. 966-972, 2017.

[11] C. Ma, K. Xue, and P. Hong, "Distributed access control with adaptive privacy preserving property for wireless sensor networks", *Security and Communication Networks*, vol. 7, no. 4, pp. 759-773, 2014.

[12] M. Mandal, G. Sharma, and A. K. Verma, "A computational review of identity-based signcryption schemes", *International Journal of Network Security*, vol. 18, no. 5, pp. 969-977, 2016.

[13] S. Movassaghi, M. Abolhasan, J. Lipman, D. Smith, and A. Jamalipour, "Wireless body area networks: a survey", *IEEE Communication Surveys & Trtorials*, vol. 16, no. 3, pp. 1658-1686, 2014.

[14] L. B. Oliveira, D. F. Aranha, et al., "TinyPBC: Pairings for authenticated identity-based non-interactive key distribution in sensor networks", *Computer Communications*, vol. 34, no. 3, pp. 485-493, 2011.

[15] Z. Shao, Y. Gao, "A provably secure signature scheme based on factoring and discrete logarithms", *Applied Mathematics & Information Sciences*, vol. 8, no. 4, pp. 1553-1558, 2014.

[16] K. A. Shim, "S2DRP: Secure implementations of distributed reprogramming protocol for wireless sensor networks", *Ad Hoc Networks*, vol. 19, pp. 1-8, 2014.

[17] K. A. Shim, Y. R. Lee, and C. M. Park, "EIBAS: An efficient identity-based broadcast authentication scheme in wireless sensor networks", *Ad Hoc Network*, vol. 11, no. 1, pp. 182-189, 2013.

[18] Y. Tian, Y. Peng, X. Peng, and H. Li, "An attribute-based encryption scheme with revocation for fine-grained access control in wireless body area networks", *International Journal of Distributed Sensor Networks*, vol. 10, pp. 713541-173541, 2014.

[19] Y. Tian, Y. Peng, G. Gao, and X. Peng, "Role-based access control for body srea networks using attribute-based encryption in cloud storage", *International Journal of Network Security*, vol. 17, no. 5, pp. 597-606, 2015.

[20] C. Zhou, G. Gao, and Z. Cui, "Certificateless signcryption in the standard model", *Wireless Personal Communications*, vol.92, no. 2, pp. 495-513, 2017.

[21] Y. Zhou, B. Yang, and W. Zhang, "Provably Secure and Efficient Certificateless Generalized Signcryption Scheme", *Chinese Journal of Computers*, vol. 39, no. 3, pp. 543-551, 2016.

Biography

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 and a lecture in Taiyuan University of Science and Technology. Her current research interests are in the area of wireless body area networks 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.

Lizhong Jin received the M.E in Computer Science and Technology in 2005 from Taiyuan University of Technology, China. He is currently a Ph.D. student at Taiyuan University of Technology. His research interests include Pattern Recognition, data mining and Big Data Analysis and Application.

A Global Intrusion Detection System using PcapSockS Sniffer and Multilayer Perceptron Classifier

Azidine Guezzaz[1], Ahmed Asimi[2], Younes Asimi[2,3], Zakariae Tbatous[2] and Yassine Sadqi[4]

(Corresponding author: Ahmed Asimi)

M2SC Team, Technology High School Essaouira, Cadi Ayyad University Marrakech[1]
Km9, Road Agadir, Essaouira Aljadida BP.383, Morocco
LaB SiV Laboratory, SCCAM Team, Faculty of Sciences, IbnZohr University[2]
Agadir, Morocco
Technology High School Guelmim ,IbnZohr University[3]
Agadir, Morocco
Km9, Road Agadir, Essaouira Aljadida BP.383, Morocco
Polydisciplinairy Faculty, Sultan Moulay Slimane University[4]
Beni Mellal, Morocco
(Email: asimiahmed2008@gmail.com)

(Received Aug. 1, 2017; Revised and Accepted Apr. 12, 2018; First Online Feb. 10, 2019)

Abstract

The evolution of networks requires a high monitoring of their resources and a reliable security of exchanges to obtain a faithful communication between their systems. The automatic detection of intrusions has become an active discipline due to the increased needs of computer security and large malicious traffic with attacks that can infect systems. Intrusion detection and prevention systems are the recent technologies used to monitor data activities. Thus, their assessment is very useful. The main goal of this paper is to analyze some sniffers tools and to assess the performances of certain intrusion detection and prevention systems. The analysis measures assess the authenticity, availability, integrity and confidentiality but also certain parameters related to security, such as: Detection type, filtering detection method, real time reaction, updating, alerting, logging. A novel detection approach is designed to perform the monitoring of networks. It is based on PcapSockS sniffer that collects data and on multilayer perceptron to analyze and make the appropriate decisions. This approach makes a reliable detection by minimization number of false positives and elimination of false negatives.

Keywords: Classification; Intrusion Detection; Performances; Security; Sniffing

1 Introduction and Notations

As long as the intrusions detection makes a network safer, prevention aims to make appropriate decisions by reacting in real time. The IDPSs (Intrusion Detection and prevention Systems) are designed for networks security needs. The sniffers tools are used to capture the circulated packets within network interfaces; they decode certain packets of a specific interest. The IDPS are used to control exchanged events through networks, to inform the existence of an intrusion, and then to take a concise action and bring systems into a safe state. The current IDPS are oriented towards automatic responses to intrusions in real time with alerts. They can be classified according to the type of detection approach, level of monitoring, frequency of use or nature of reaction. False positives are generated when a detection system identifies normal activity as an intrusion, while false negatives correspond to undetected intrusions, so no alert is generated. It is impossible to find a standard detection tool that can overcome all limitations. The second section presents a state of art on intrusion detection, sniffing and multilayer perceptron. The performances analysis is cited in the third part, based on security objectives and on parameters related to security. For the fourth section, the proposed solutions are described. The article is accomplished by a conclusion and the future works. In this paper, we use the following notations (Table 1):

Table 1: Notations

f	: Sigmoid Function.
$(X_i)_{i=1..n}$	: The presented inputs.
$X_i = (x_{i,j})_{j=1..m}$	: The presented occurrences to input X_i.
$W^{(0)} = (w_{i,0})_{i=1..n}$	: The initialized weights.
$W_i = (w_{i,j})_{j=1..m}$	: The associated weights to input X_i.
$w_{0,i}$	: Initialized Bias to 1 and associated to input X_i.
a_i	: Weighted sum associated to input X_i.
$y(a_i) = f(a_i)$	: Calculated output associated to input X_i.
ϵ_i	: Calculated error associated to an entry X_i.
$W_i^{op} = (w_{i,j}^{(op)})_{j=1..m}$	: Optimal system solution (Training Algorithm) for X_i.
$W_{0,i}^{op}$	: Optimal System Bias (Training Algorithm) for X_i.
$a_i^{(op)}$	: Optimal weighted sum associated to an input X_i.
$a_i^{(max)}$	: Maximum weighted sum associated to an input X_i.
$W^{max} = (w_j^{(max)})_{j=1..m}$	: Maximum weights.
$w_0^{(max)}$	: Maximum bias.
$d = +1$	: Normal Output.
$d = -1$	: Anormal Output.

2 State of Art

This section gives a state of art of IDPS, the sniffing techniques and multilayer perceptron.

2.1 Intrusion Detection and Prevention

Intrusion detection is a set of techniques used to detect undesirable activities. An intrusion attempts to violate one of security objectives [2, 11]. An IDPS can be software or hardware which can detect malicious events that attempt to infect a security policy. IPS (Intrusion Prevention Systems) are considered as second generation detection systems, designed to make necessary decisions to stop the detected intrusions accurately. There are two fundamental detection methods [2, 7, 11, 12, 17, 18, 24]:

- Scenario approach that identifies an intrusion using a configuration known for malicious activity.
- Behavioral approach that attempts to identify malicious based on a deviation from normal activity. It is proposed by J. P. Anderson (1980) and extended by D. E. Denning (1987).

An IDPS can control and detect accurately the abnormal activities by blocking them quickly. It is characterized by following properties:

- Real time takes into account time constraints and delays related to the results.
- Response time which determines the duration between activation and time of the results.
- Blocking is used to interrupt the passage of suspicious activities.
- Alert is a message generated after detection to inform the manager about the existence of an intrusion.

The IDPS architecture is composed by [10,12,15,16,24,28] (Figure 1):

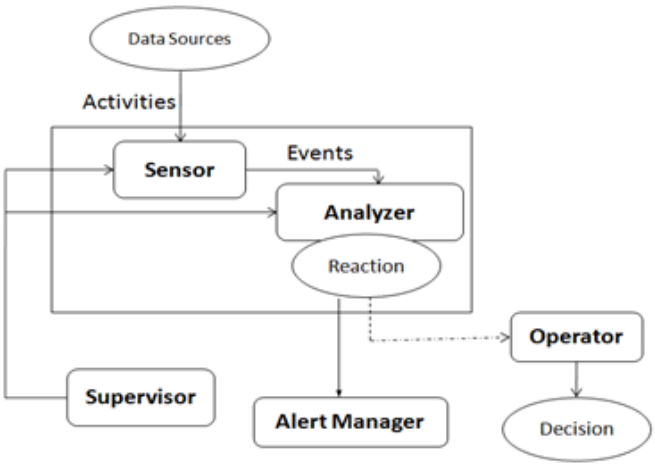


Figure 1: Classical architecture of IDPS

- Data sources contain the data that reflects what is happening on the hosts and the traffic of packets that is intercepted by a network monitor.
- Activities are collected within data sources and stored in database.
- Sensor observes the system activities through data sources and provides a sequence of events that inform evolution of the system state.

- Events represent the preprocessed activities presented to analyzer.
- Analyzer determines if the events contain malicious activities.
- Reaction is guaranteed by activating the countermeasures to end the detected attack.
- Supervisor is responsible to analyze alerts and has a global vision toward system.
- Alert manager is used to generate alerts after detection.
- Operator is a part of IDPS that make a final decision.

The majority of current IDPSs integrate heterogeneous technologies like, VPN (Virtual Private Network), antivirus, antispam, *etc.*

2.2 Sniffing Techniques

The sniffing is a process used to intercept and analyze the network traffic. It listens to public conversations in computer networks [6, 14, 27]. The sniffers may be used as a hardware or software solution or as software only to manage and ensure the network security. They can also be used by unauthorized uses. An intruder can learn network configuration information by sniffing. There are different types of sniffing packets [21, 22]:

- IP (Internet Protocol) sniffing collects all IP packets traveled through a network.
- MAC (Medium Access Control) sniffing captures the corresponding frames to supervised interfaces MAC addresses.
- ARP (Address Resolution Protocol) sniffing intercepts ARP packets used to query the ARP cache.

The sniffers are constituted by the components described by Clincy & Abi Halaweh in [3, 27]:

- Hardware is represented by Network Interface Cards and activated in sniffing mode.
- Driver captures data from the network cards, applies a number of filters and stores it in a memory.
- Buffer stores the captured traffic or transfers it to permanent storage.
- Analyzer is responsible to analyze the traffic in real time taking into account the criteria needs.
- Decoder receives a stream of bits and interprets them to finally build a descriptive texts format.
- Editor changes the traffic using a unified format and then converts it and retransmits it in the network.

The filtering is an essential operation to classify packets that are captured using filters according to the needs of capture. The simulation with sniffing tools is used in learning of computer networking, allows a good understanding of network concepts and topologies. The study carried on [14] highlights the difference between these two principles libraries. The main capture libraries are libnet and libpcap. The Airpcap adapter is used on hosts running to listen in to wireless traffic in monitor mode.

2.3 Multilayer Percepron

The birth of artificial neural discipline dates back to the 1940s with W. McCulloch and W. Pitts who showed that with such networks, we could in principle calculate any arithmetic or logical function [23]. The training is a dynamic and iterative process [29] used to modify the parameters of network in reaction with stimuli that receives from its environment. The supervised training adjusts network parameters by a direct comparison between the actual network output and the desired output. The unsupervised training involves no target values. It tries to associate information from the inputs with an intrinsic reduction of data dimensionality or total amount of input data. The type of training is determined by how the parameter changes occur [16, 26]. The MLP (Multilayer perceptron) (Rosenblatt 1957) is a neural network that composed of successive feedforward layers connecting neurons by weighted links [15, 16, 29]. The input layer is used to collect the input signals and the output layer provides responses. One or more hidden layers are added for transfer. The training of MLP is performed by the error gradient propagation. In the 1980s, an error propagation algorithm was invented [29]:

Algorithm 1: Back propagation training

1) DBA : Training Base.
   $X_i = (x_{i,j})_{j=1...m}$: Inputs.
   $C_i = (c_{i,j})_{j=1...m}$: Desired Results for X_i.
   $W_i = (w_{i,j})_{j=1...m}$:Weights for X_i.
   θ_i: Calculated Results.
   λ_i:Training rate.

2) BEGIN : Calculate W_i for the input X_i

   For i from 1 to n do
   Initialize the weights randomly
   Optimization of weigts:
   For j from 1 to m do
   $w_{i,j} = w_{i,j-1} + \lambda_i(c_{i,j} - \theta_i)x_{i,j}$
   EndFor
   EndFor

3) END

The examples of the training basis are shown successively in order to adjust the weights by accumulating the

calculated gradients. The training is stopped when the calculated error is less than a certain threshold.

3 Our Contribution

In this section, we describe the results of performances analysis carried on some network sniffers and certain IDPS. It proposes a novel model of IDPS based on PcapSockS sniffer and multilayer perceptron.

3.1 Results of Performances Analysis

The sniffers analyze data from all the network layers. If the application level analysis fails to identify the problem and find a solution, sniffers can dig into lower level details. Based on various criteria and referring to the detailed study in [13, 14, 21, 27], we arrive at a classification of the following systems (Table 2):

After this assessment, the majority of sniffers above use libpcap library to intercept traffic and include a filtering system. They are highly available to monitor wired and wireless networks with a high flows supporting a large number of protocols. The study helps us to discover certain limitations. The actual sniffers are more efficient, allowing real time analysis. They capture packets from the network and decode them into human readable format. To be able to choose a better detection system before installing it on the affected network, it is useful to test and evaluate the operational efficiency of these systems.

- Snort is an open source network IDPS, developed by Sourcefire. It is a scenario and anomaly system [8, 9, 25].
- Suricata is an open source IDPS that uses the snort rules, its important advantage is multithreading that means reduction of time and gives also a high performances. David and Benjamin analyzed Snort and Suricata and conclude that Suricata is relevant and exact than Snort [8, 11, 25].
- Mc Afee Host Intrusion Prevention is aiming to protect systems, resources and applications. It establishes reporting and gives an exact management, progressed and easy to use [11, 16].
- Net ASQ is an engine integrating intrusion prevention and eliminates intrusions in real time. Its hardware alternative arranges a Watchdog which realizes regularly tests of activities [11, 16].

To satisfy this assessment, we propose, the degree of guarantee of the safety objectives: authenticity, confidentiality, integrity, availability [11, 16, 20] (Table 3):

Most of the existing solutions concerning intrusion detection are related to the setting up of NIDPS in association with some HIDPS and other software types of management. It has been observed that NIDS become less effective even when presented with a bandwidth of a few hundred megabits per second.

3.2 Our Proposed Approach

This proposition is based to avoid some vulnerabilities and limits. The structure of system is (Figure 2):

Our IDPS system is constituted by the different components bellow:

- Data sources: The circulated data flow within the network are intercepted and processed to monitor and make an effective decision.
  - ∗ High level means the monitoring of various activities within the high layers.
  - ∗ Low level means the monitoring of various activities within the low layers.
- Sensor observes the data and provides the analyzer a sequence of activities that inform the evolution of the system state.
  - ∗ PcapSockS Sniffer intercepts traffic from the low and high level.
  - ∗ Activities: the collected data are stored in a collection base in the form of activities.
- Analyzer is made to take a decision by exploiting the implemented detection methods.
  - ∗ Normalization: is located directly after sniffing which is used to eliminate the potential ambiguities and to have a uniform structure of activities.
  - ∗ Comparator is a component that compares an event with the contents of the intrusion basis.
  - ∗ Events: the normalized activities become events and presented to analyzer.
  - ∗ Multilayer Perceptron Classifier is able to distinguish the normal behavior from the new data.
  - ∗ Notification: after detection of intrusion the analyzer sends notification to manager.
  - ∗ Updating: the intrusions basis is updated in order to increase the possibilities of the new detections and to facilitate the next analysis.
- Manager is responsible for the management and analysis of the alerts generated by the analyzer. It contains:
  - ∗ Management: the manager is responsible to analyze alerts and take action to prevent the damage of intrusion.
  - ∗ Real time blocking means the realtime response to block intrusion and anticipate connection.
  - ∗ Automatic reaction provides reaction mechanisms to cope with detected intrusion or reduce their effect.
- Supervisor is the person who administers the various components of that system. He has a global vision on the system.

Table 2: Performances assessment of some network sniffers

Network sniffers	S/H	Library	Filtering	Flow	Availability	Alert	Real time
Tcpdump	S	Libpcap (Winpcap)	++	Flow of Ethernet networks	Very economical installation file size: 484 KB	- -	- -
Wireshark	S	Libpcap (Winpcap)	++	Flow of Ethernet and wireless networks.	81 MB after installation.	- -	++
PACKETYZER	S	Libpcap (Winpcap)	++	Flow of Ethernet, FDDI, PPP, Token Ring and wireless networks.	-supports 483 protocols. -Decodes and edits packets.	- -	++
Netflow CISCO	S H	Libpcap	++	High flow networks (Gigabit).	Very high (provides valuable information about users, network applications, peak hours). -2GH Dual processor. -2GO Memory.	++	++
Colasoft Capsa	S	Libpcap	++	Flow wired and wireless networks over 802.11a, 802.22b, 802.11g and 802.11n	-No Tolerant with the attacks: ARP, TCP port scanning, -Signals DOS attacks -653 MB on after windows 7 installation. - Free version is available with limited features.	++	++
PRTG Network Monitor	S H	Libpcap	++	High flow	-Integrates SNMP, Packet (Sniffing and Net flow). -monitors 24/7 network. - Includes over 200 types of sensors. -Less than 30 protocols (Free). - More than 30 protocols (Com)	++	++
Kismit	S	Libpcap	++	Flows of wireless networks 802.11n, 802.22b 802.11g and 802.11a	High (supports any wireless card rfmon)	++	++
Scapy	S	Libpcap and Libnet	++	Injectes the 802 frames	- Generates and receives quick and accurate traffic. - Decodes packets of a number of protocols.	++	++
OmniPeek	S H	Libpcap	++	Ethernet, Gigabit, 10 Gigabit, 208.11 a / b / g / n / ac wireless, VoIP, Video, MPLS and VLAN	-captures on multiple networks simultaneously. - Several hundred protocols - WPA, WPA2 and PSK Decoding.	++	++
ETHERAP	S	Libpcap	++	Flows of Ethernet, FDDI, Token Ring, ISDN.	- Is only available for GNU / Linux systems.	- -	++
Soft Perfect Network Protocol Analyzer	S	Libpcap	++	Flows of Ethernet networks	-Analyzes of fragmented floors. -Defragments and reassembles the packets. - Size of the installation file 4.87 Mb.	- -	++
Airodump	S	Libpcap	++	- Wireless networks 802.11. - Supports 4.2 GHz channels	-Identification the coordinated access points. -Writes the several files containing details of all seen access points and clients.	- -	++

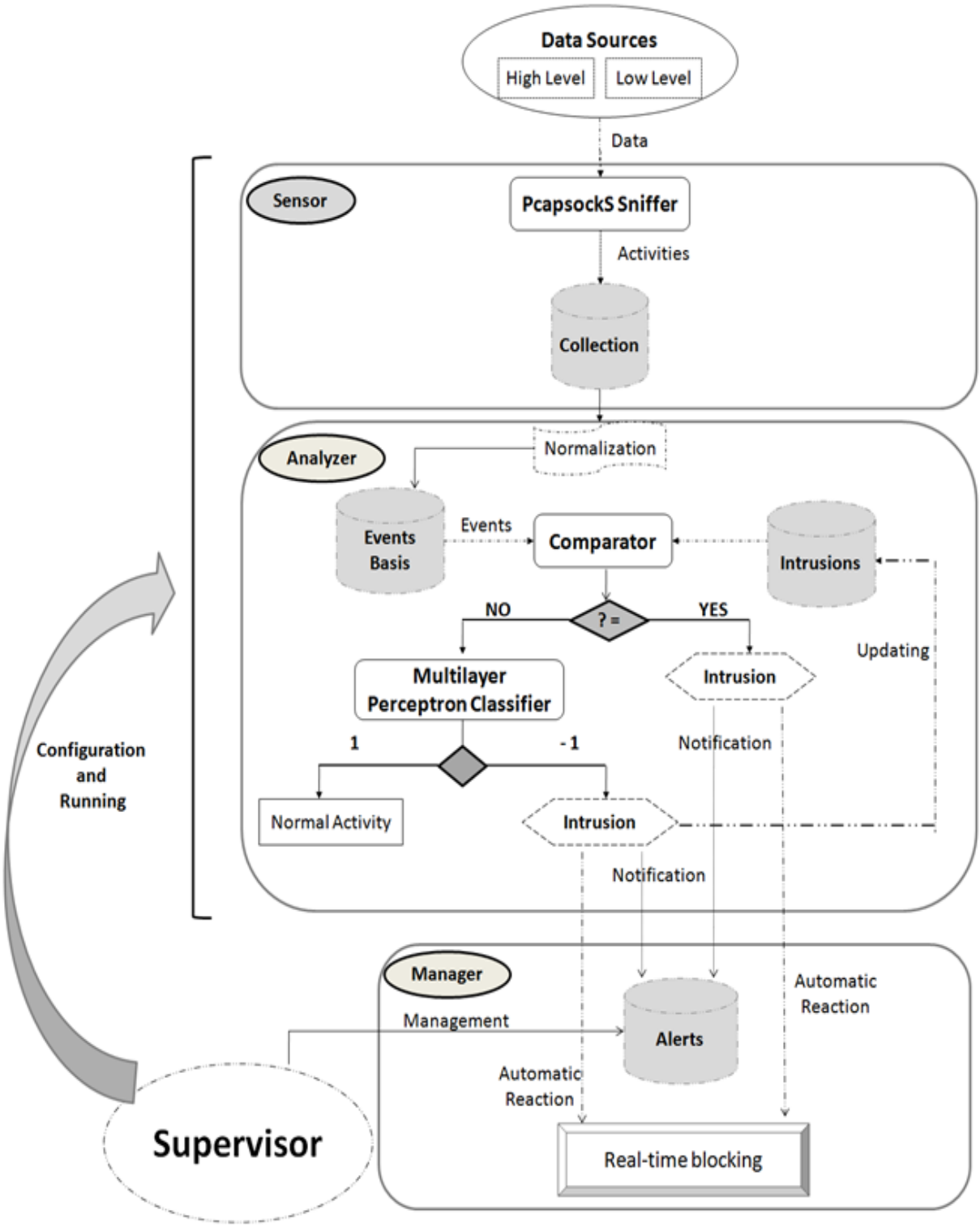


Figure 2: Proposed IDPS based on PcapSockS sniffer and MLP classifier

Table 3: Assessment based on security objectives

	Snort	NetASQ	Suricata	Winpooch	MC Afee Intercept	Bro	Net Screen Intrusion	Cisco Net Ranger
Authenticity	High	Protocole IPSEC, Certificats X.509, PKI infrastructures, SSL	High	MAC algorithm	High	High	Medium	ACL Lists
Confidentiality	TowFish Algorithm	DES, 3DES, AES, BlowFish.	Cryptografic functions of TLS protocol	——	——	Include SSH functions	RC4 Algorithm	——
Integity	5MB/s, 10MB/s, 4GB/s	High speed MD5, SHA1, SHA2	Hush functions of TLS protocol	Includes scan of ClamWin antuvirus	——	high-level semantic analysis/ detect a large number of protocols	5MB/s, 100MB/s, 1GB/s/	High reliability
Availability	Continuous frequency	Continuous frequency	Continuous frequency	High	Continuous frequency	Continuous frequency	Continuous frequency	Continuous frequency

The use and management of databases is very important in this approach; we opted for using of four databases:

- Collection basis is composed by activities that intercepted within networks by PcapsockS sniffer.
- Events basis is constituted by the normalized activities.
- Intrusions basis includes all known attacks by using a certain format. There is no standard for the coding of attacks. It is updated after detection.
- Alerts basis contains different alerts generated after detection by our IDPS.

Our IDPS performs the first monitoring based on signature detection. Thus, it needs signatures basis that will satisfy this type of detection. Therefore, we have to conceive intrusions basis which characterize the anomalies of the monitored network (Figure 3).

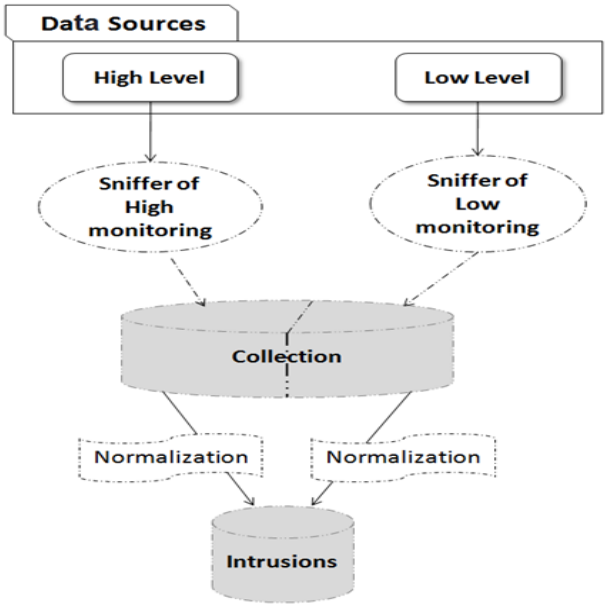


Figure 3: Conception of intrusions basis

The monitoring is done in a hybrid and complete way by controlling all levels of data sources. For this, we use the most famous sniffing tools more used currently to meet our needs. For example, we use Scapy [13,14,16] for high level sniffing and Wireshark [3,14,21] for the lower layers. The collected activities are recorded in a collection basis. In this case, the sniffing type takes place during so called abnormal operation of the network to collect abnormal activities characterizing the anomalies of the monitored network. The intrusions are used by detection systems. They are stored and integrated into their database at each infection. The detection is carried out by comparing the event collected the contents of the intrusion database. To implement the new approach, various phases are used:

3.2.1 Collection and Filtering Phase

The proposed design in [1] focuses on the combination of current performances of high sniffers and minimization of various limitations. It is a distributed model consisted by two main components:

- The kernel is composed by two processors to capture and filter the traffic.
- The operator decodes and normalizes the elected traffic.

These components are described in the figure below (Figure 4):

This traffic is composed of a set of bits and frames, it's saved in a temporary basis to apply the BPF (Berkeley Packet Filter) and then meet adequate collection conditions. Libpcap provides the possibility to introduce the filters to filter traffic: PBF, SWIF. It applies the filters on traffic in the basis in order to choose the elected packets. This latter is redirected to the operator space. The decoding processor normalizes and stores the chosen traffic in the collection Database. In the high level, we use the sockets mechanism to ensure a reliable collection. The TCP (Transmission Control Protocol) and UDP (User Datagram Protocol) sockets are implemented for this purpose. Raw sockets are used to reinforce the interception to the low level with libpcap. The collected traffic is saved in a temporary basis to apply the filter LSF and redirected directly to Collection Database. Our sniffer collects data in three modes:

- Connection oriented mode requires a prior connection establishment between communicating entities.
- Connectionless mode cannot guarantee a reliable connection, insertion errors, wrong delivery, duplication, or non sequencing delivery packets.
- Raw mode can provide both services in connection oriented and connectionless mode.

The filtering provides a considerable gain; it avoids the congestion and the saturation of memory. The filtering is a very useful to meet the various network services using mainly in intrusion detection. The treatments are in real time. Take into account the time constraints which are as important as the accuracy of the results for this system synchronizes multiple tasks that take place and the possibility of including several shorter threads in a single process. To show the performances provided by PcapSockS Sniffer, it is very useful to compare it with other sniffers which have demonstrated their reliability (Table 4):

The new model combines libpcap and sockets functions to capture the packets, filters traffic taking into account the capture needs. All treatments are in real-time and Encryption of transactions between the sniffer and Collection database.

3.2.2 Preprocessing and Normalization Phase

The preprocessing phase is the most labor intensive, due in particular to the lack of structuring and the large amount of noise existing in the raw data used. It consists in structuring the activities in order to prepare them for a future analysis. The significant formatting is required before analyzing and classifying traffic. The normalization is carried out also to establish a pattern of activities facilitating the distinction between the activities and allowing an extraction of useful fields if necessary. The events are stored in a database table which contains columns to specify the fields and contains the occurrences with string type. A hash function is applied to compute the signatures of the content. We realize a particular coding for the enumeration of the occurrences and adapt them to the entries of the model which accept in principle only integer, real or Boolean entries. The hashing and coding techniques guarantee also a certain rapidity and integrity. The input layer receives successively the preprocessed occurrences. An occurrence is subdivided into a set of fields. Each field is received by a neuron representing a simple receptor that does not perform any treatment. A weighted sum is calculated on the input values. A transfer function is applied to the calculated sum. The sigmoïde function is implemented in the hidden layer.

3.2.3 Classification Phase

We propose a rigorous algorithm for training and recognition. Thus, we use the multilayer perceptron. The proposed classifier is [15, 16, 19] (Figure 5).

Each layer has neurons directly linked to the neurons of the next layer. We find ourselves faced with an optimization model containing changeable variables that describe the problem, together with constraints representing limits on these variables. We define a cost function to minimize is:

$$a_i = \sum_{j=1}^{m} w_{i,j} x_{i,j} + w_{0,i} \text{ for } i = 1, \cdots, n.$$

The inputs preprocessing are used to remove redundant and irrelevant information in order to achieve a small and optimal network structure.

Algorithm 2: Training algorithm
Initialize weights $W^{(0)} = (w_{i,0})_{i=1\ldots n}$ such as $w_{i,0} \leqslant 10^{-3}$ for $i = 1 \ldots n$ and $w_{0,i} = 1$.
For i from 1 to n do

1) Present the iutputs $X_i = (x_{i,j})_{j=1\ldots m}$.
2) Calculate $W_i^{(op)}$ and ϵ_i :
$\epsilon_i = \min_{a_i}(1 - y(a_i))$

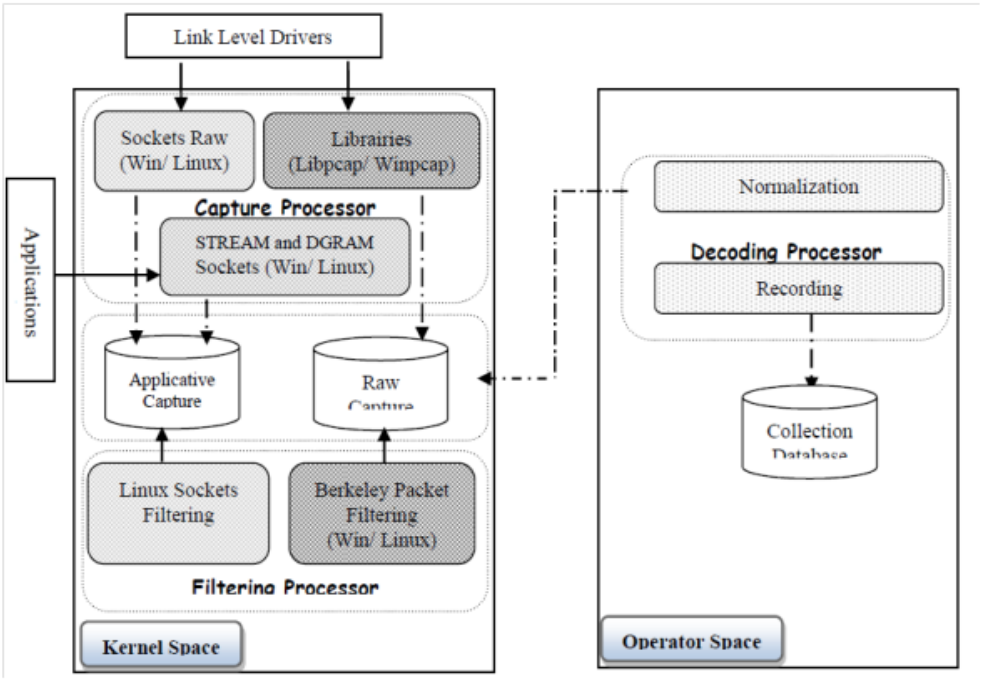


Figure 4: The pcapSockS sniffer

Table 4: Comparison between pcapSockS sniffer, scapy and wireshark

Sniffer	Platforms	Low capture	High capture	Low filtering	High filtering	Network
Scapy	-Win -Mac OS	-Libpcap -Linux	-Libnet	-PBFfilter -Python Functions	-No	-Wired -Wireless
Wireshark	-Win -Linux	-Libpcap	-No	-PBF Filter	-No	-Wired -Wireless
Pcap.Sock Sniffer	-Win -Linux	-Libpcap -Raw Sockets	- Sock_Stream -Sock_Dgram	-PBF Filter	-LSF Filter	-Wired

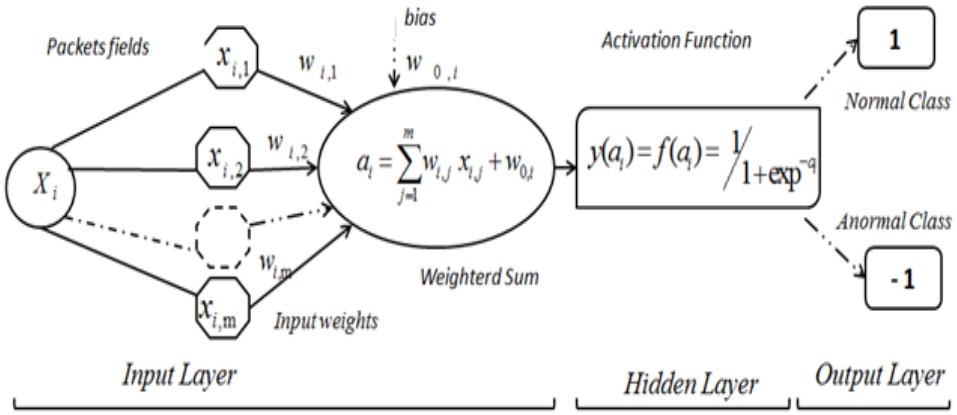


Figure 5: Multilayer perceptron classifier

$$\left\{\begin{array}{l} a_i = \sum_{j=1}^{m} w_{i,j}x_{i,j} + w_{0,i}; \\ y(a_i) = f(a_i); \\ w_{0,i} = w_{0,i} + [1 - y(a_i)]; \\ \text{For } j \text{ from 1 to } m \text{ do} \\ \quad w_{i,j} = w_{i,j-1} + [1 - y(a_i)]x_{i,j}; \\ \text{EndFor} \end{array}\right\}$$

3) EndFor

In the following, we denote with:

$W^{(max)} = (w_j^{(max)})_{j=1\ldots m}$ with $w_j^{(max)} = max\{w_{i,j}^{(op)}, i = 1\ldots n\}$ and $w_0^{(max)} = max\{w_{0,i}^{(op)}, i = 1\ldots n\}$, $a_i^{(max)} = \sum_{j=1}^{m} w_{i,j}^{(max)}x_{i,j} + w_{0,i}^{(max)}$ $a_i^{(op)} = \sum_{j=1}^{m} w_{i,j}^{(op)}x_{i,j} + w_{0,i}^{(op)}$ and $S = \{(X_{i=1}^{n} = (x_{i,j})_{j=1\ldots m}, W_i^{(op)} = (w_{i,j}^{(op)})_{j=1\ldots m}, w_{0,i}^{(op)}, \epsilon_i), i = 1\ldots n\}$ the obtained results during the training phase.

Proposition 3.1. *With the above assumptions, we then have for all* $i \in \{1, \ldots, n\}$

1) $a_i^{(max)} \geq a_i^{(op)}$.

2) $0 < 1 - y(a_i^{(max)}) \leq \epsilon_i$.

Proof.

1) As $w_j^{max} = max\{w_{i,j}^{(op)}, i = 1\ldots n\} \geq w_{i,j}^{(op)}$ and $x_{i,j} \geq 0$ for all $i = 1\ldots n$, then $a_i^{(max)} \geqslant a_i^{(op)}$ for all $i \in \{1, \ldots, n\}$.

2) We have $1 - y(a_i^{(op)}) = \epsilon_i$ for each $i \in \{1, \ldots, n\}$, $y(a_i^{(op)}) = f(a_i^{(op)})$. Therefore $0 \leqslant f(a_i^{(op)}) \leqslant f(a_i^{(max)}) < 1$ for each $i \in \{1, \ldots, n\}$ because f is an increasing function.

   Thereafter $\epsilon_i = 1 - y(a_i^{(op)}) = 1 - f(a_i^{(op)}) \geqslant 1 - f(a_i^{(max)}) = 1 - y(a_i^{(max)}) > 0$

   which shows (2).

□

This phase consists of validating the model: We use for this the optimized weights which are obtained during the training phase.

Definition 3.1. *Let* $K = (k_j)_{j=1\ldots m}$ *be an input occurrence and* $a = \sum_{j=1}^{m} w_j^{(max)}k_j + w_0^{(max)}$.

1) K is a normal occurrence if there exits $i \in \{1, \ldots, n\}$ *such that* $1 - y(a) \leqslant \epsilon_i$.

2) K is an intrusion occurrence if for all $i \in \{1, \ldots, n\}$ *we get* $1 - y(a) > \epsilon_i$.

Proposition 3.2. *Let* $K = (k_j)_{j=1\ldots m}$ *be an input occurrence,* $a = \sum_{j=1}^{m} w_j^{(max)}k_j + w_0^{(max)}$ *and* $\epsilon = \{\epsilon_i, i = 1, \ldots, n\}$.
The following conditions are equivalent:

1) K is an intrusion.

2) $1 - y(a) > \epsilon$.

The proof of this proposition relies on Definition 3.1.

Corollary 3.1. *: Let* $K = (k_j)_{j=1\ldots m}$ *be an input occurrence,* $a = \sum_{j=1}^{m} w_j^{(max)}k_j + w_0^{(max)}$ *and* $\epsilon = max\{\epsilon_i, i = 1, \ldots, n\}$. *The following conditions are equivalent:*

1) K is a normal information.

2) $1 - y(a) \leq \epsilon$.

The proof of this corollary relies on Definition 3.1 and Proposition 3.2.

Algorithm 5: Recognition algorithm

1) New input $X = (x^{(j)})_{j=1\ldots m}$, final output d, activation state a, calculated result $y(a)$.

2) Computing of output
   $a = \sum_{j=1}^{m} w_j^{(max)}x^{(j)} + w_0^{(max)}$;
   $y(a) = f(a)$;

3) Classification of activities

$$\left\{\begin{array}{l} if\ (1 - y(a) \leqslant \epsilon)\ then \\ \quad d = 1\ //Normal\ activity \\ \quad else \\ \quad d = -1\ //Intrusion \\ Endif \end{array}\right\}$$

The sigmoid is introduced into the two proposed algorithms for training and recognition. It presents certain constraints during its implementation which leads us to make an evaluation of the sigmoid on platforms more used in practice. In this case, we determine the random values of the weights in $w_i \leqslant 10^{-3}$ to ensure the possible results and avoid the falsified outputs. This modeling leads us to develop an optimal and restricted database containing the occurrences (Table 5):

Table 5: Database structure

$W^{(max)} = (w_j^{(max)})_{j=0}^{m}$	ϵ

3.3 Evaluation Study of Proposed Approach

To accomplish such evaluation, a set of measures are available: precision, number of false positives and false negatives. We take into account the parameters cited in [5,11,14,16,20] such as data sources, intrusion response, frequency of use, real time analysis, intrusion blocking method, real time alert, logging, filtering methods, compatible operating systems (Table 6).

The proposed classifier defines a supervised method using a three layer perceptron and represents a perceptual analysis layer which can be integrated into our new detection system to monitor traffic and make control of the data flow more reliable. With a series of experimental studies on a set of intrusion detection and prevention systems much responded actually. We demonstrated that our new proposal model is more efficient, especially at the level, of data collection, pretreatment of the activities and their classification.

Table 6: Assessments based on proposed parameters

	Snort	**NetASQ**	**Suricata**	**MC Afee Intercept**	**Bro**	**Cisco Net Ranger**	**Our Proposed IDPS**
Realtim Analysis	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Type	Nips/ Hips	Nips/ Hips	Nips	Nips	Nips	Nips	Nidps/ Hidps
Detection type	Signature	Signature/ Behavioral	Signature/ Behavioral	Signature/ Behavioral	Signature/ Behavioral	Signature	Signature/ Behavioral
Blocking	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Reaction	Active	Active	Active	Active	Passive	Active	Active
Real time Alerts	Yes	Yes	Yes	Yes/ Email, S NMP	Yes	Yes	Yes
Logging	Yes	Yes	Yes	Yes/ MS SQL Server	Yes	Yes	Yes
Filtering	Determined by Admin	Filtering and flow contol	Yes	Yes	Yes	Yes	Yes
Updating	Manual	Automatic	Manual	—	Automatic / Manual	Automatic	Automatic
Operating Systems	Linux/ Windows	Linux/ Windows	OS Commun	Windows/ Solaris	Linux/ Free BSD/ MAC OS	OS Commun	Linux/ Windows
Modularity of components	No	No	No	No	No	No	Yes
Data collection	Medium	Medium	Medium	Medium	Medium	Medium	High
Classification of atvities	Medium	Medium	Medium	Medium	Medium	Medium	High

4 Conclusion

An IDPS tries to detect malicious activities and attacks. It attempts to control computers by monitoring traffic. Many methods have been employed. In this paper, we perform performances evaluation of a list of sniffing and intrusion detection tools and we deduct in the end that those tools suffer much vulnerabilities. So, we propose an optimal approach of intrusion detection based on multilayer perceptron technique aiming to improve the accuracy of detection. The modeled system aims to protect networks from attacks on service integrity, authentication and confidentiality. The preprocessing is performed to transform data evens into a new representation before being presented to a neural network inputs. With implementations carried on different parts, we demonstrate that our system gives the solutions more reliable and relevant to improve the o the network security. Our next work will focus and discuss in detail the various steps of implementation and validation of this global system describing the proposed solutions.

References

[1] Amrita, K. Ravulakollu, "A hybrid intrusion detection system: Integrating hybrid feature selection approach with heterogeneous ensemble of intelligent classifiers," *International Journal of Network Security (IJNS'18)*, vol. 20, no. 1, pp. 41-55, 2018.

[2] V. Alanou, Détection D'intrusion Dans Un Système Informatique : Méthodes Et Outils, 1996. (`http://www.secdev.org/idsbiblio/lme96methodes.pdf`)

[3] Asrodia, *et al.*, "Analysis of various packet sniffing tools for network monitoring and analysis," *International Journal of Electrical, Electronics and Computer Engineering*, vol. 1, no. 1, pp. 55-58, 2012.

[4] J. Balasubramaniyan, J. O. Garcia-Fernandez, D. Isacoff, E. Spafford, D. Zamboni, "An architecture for intrusion detection using autonomous agents," *COAST Laboratory Purdue University West Lafayette*, 1998. ISBN: 0-8186-8789-4.

[5] M. Boujnouni and M. Jedra, "New intrusion detection system based on support vector domain description with information gain metric," *International Journal of Network Security (IJNS'18)*, vol. 20, no. 1, pp. 25-34.

[6] L. Chappell, "Wirehark 101 essential skills for network analysis," *Protocol Analysis Institute, Inc*, 2013. (https://www.wiresharkbook.com/101v2_samplepages/Wireshark978-1893939752-toc.pdf)

[7] A. Chaudhary, V. N. Tiwari, and A. Kumar, "A New intrusion detection system based on soft computing techniques using neuro-fuzzy classifier for packet dropping attack in MANETs," *International Journal of Network Security*, vol. 18, no. 3, pp. 514-522, 2017.

[8] D. J. Day, B. Burns, "A performance analysis of snort and suricata network intrusion detection and prevention engines," in *The Fifth International Conference on Digital Society*, 2011. (https://www.researchgate.net/publication/241701294_A_Performance_Analysis_of_Snort_and_Suricata_Network_Intrusion_Detection_and_Prevention_Engines)

[9] O. Eldow, P. Chauhan, P. Lalwani, M. Potdar, "Computer network security ids tools and techniques (snort/suricata)," *International Journal of Scientific and Research Publications*, vol. 6, no. 1, pp. 593, 2016.

[10] Y. Farhaoui, A. Asimi, "Creating a complete model of an intrusion detection system effective on the LAN," *International Journal of Advanced Computer Science and Applications (IJACSA'12)*, vol. 3, no. 5, 2012.

[11] Y. Farhaoui and A. Asimi, "Performance method of assessment of the intrusion detection and prevention systems," *International Journal of Engineering Science and Technology (IJEST'11)*, vol. 3, 2011. ISSN: 0975-5462.

[12] Y. Farhaoui, "Design and implementation of an intrusion prevention system," *International Journal of Network Security*, vol. 19, no. 5, pp. 675-683, 2017.

[13] C. Gandhi, G. Suri, R. P. Golyan, P. Saxena, B. K. Saxena, "Packet sniffer – A comparative study," *International Journal of Computer Networks and Communications Security*, vol. 2, no. 5, pp. 179–187, 2014.

[14] A. Guezzaz, *et al.*, "A new hybrid network sniffer model based on Pcap language and sockets (PcapSockS)," *International Journal of Advanced Computer Science and Applications (IJACSA'16)*, vol. 7, no. 2, 2016.

[15] A. Guezzaz, *et al.*, "A novel scheme of an intrusion system using multilayer perceptron," *The Second International Day on Computer Science & Applied Mathematics in Saculty of Sciences and Techniques (ICSAM'17)*, vol. 9, no. 4, 2017.

[16] A. Guezzaz, *et al.*, "A hybrid NIPS based on pcapSockS sniffer and neural MLP," in *International Conference on Information Technology and Communication Systems in National School of Applied Sciences (ITCS'17)*, pp. 253-266, 2017.

[17] A. Guezzaz, *et al.*, "A lightweight neural classifier for behavioral detection," in *International Conference on Information Technology and Communication Systems in Faculty of Sciences and Techniques (IWAM'17)*, vol. 2, no. 2, pp. 57-66, 2017.

[18] A. Gupta, M. Kumar, A. Rangra, V. K. Tiwari, P. Saxena, "Network intrusion detection types and analysis of their tools," *Department of Computer Science and Information Technology, Jaypee University of Information Technology*, vol. 2, no. 1, pp. 63-69, 2012.

[19] A. Guezzaz, *et al.*, "A lightweight neural classifier for intrusion detection," *General Letters in Mathematics*, vol. 2, pp.57-66, 2017.

[20] A. Guezzaz, *et al.*, "A hybrid NIPS based on PcapSockS sniffer and neural MLP," *International Conference on Information Technology and Communication Systems (ITCS'17)*, pp. 253-266, 2017.

[21] A. Halaweh,"A taxonomy of free network snifers for teaching and research," *Journal of Computing Sciences in Colleges*, vol. 21, no. 1, pp. 64-75, 2005.

[22] B. Ma, "Using packet sniffing to teach networking concepts," *Journal of Computing Sciences in Colleges*, vol. 30, no. 6, pp. 67-74, 2015.

[23] N. Malik, "Artificial neural networks and their applications," *National Conference on Unearthing Technological Developments & their Transfer for Serving Masses GLA ITM*, 2005. (https://www.researchgate.net/publication/1958135_Artificial_Neural_Networks_and_their_Applications)

[24] L. Moulad, H. Belhadaoui, M. Rifi, "Implementation of a hierarchical hybrid intrusion detection mechanism in wireless sensors network", *International Journal of Advanced Computer Science and Applications (IJACSA'17)*, vol. 8, no. 10, 2017.

[25] M. Ridho, F. Yasin, M. Eng, "Analysis and evaluation snort, bro, and suricata as intrusion detection system based on linux server," *Department of Informatics, Faculty of Communications and Informatics Universitas Muhammadiyah Surakarta*, 2014. (http://etd.eprints.ums.ac.id/31281/)

[26] M. Rochaa, *et al.*, "Evolution of neural networks for classification and regression," *Neurocomputing*, vol. 70, no. 16-18, pp. 2809-2816, 2007.

[27] Rupam, A. Verma, A. Singh, "An approach to detect packets using packet sniffing," *International Journal of Computer Science & Engineering Survey (IJCSES'13)*, vol. 4, no. 3, 2013.

[28] B. Santos, T. Chandra, M. Ratnakar, S. Baba, N. Sudhakar, "Intrusion detection system types and prevention," *International Journal of Computer Science and Information Technologies*, vol. 4, no. 1, 2013.

[29] Schiffmann, *et al.*, *Optimization of the Backpropagation Algorithm for Training Multilayer Perceptrons*, 1994. (`http://www.cs.bham.ac.uk/~pxt/NC/schiffmann.bp.pdf`)

Biography

Guezzaz Azidine received his Ph.D in design and validation of an intrusion detection and prevention system based on neural network for cloud computing. His research interest is Intrusion Detection and Prevention and Computer and Network Security and Cryptography. He is an assistant professor at the Technology High School Essaouira Cadi Ayyad University Marrakech, Morocco.

ASIMI Ahmed received his PhD degree in Number theory from the University Mohammed V – Agdal in 2001. He is reviewer at the International Journal of Network Security (IJNS). His research interest includes Number theory, Code theory, and Computer Cryptology and Security. He is a full professor at the Faculty of Science at Agadir Morocco since 2008.

Younes Asimi received his Ph.D. in Strong Zero-Knowledge Authentication Based on virtual passwords per session and the Session Keys in 2015. His research interests include Authentication Protocols, Computer and Network Security and Cryptography. He is an assistant professor at the Technology High School Guelmim, Ibn-Zohr University Agadir, Morocco.

Tbatou Zakariae received his Ph.D degree in. He is currently Ph.D student in Authentication Protocols Kerberos for distributed systems. His research interests include Authentication Protocols, distributed systems, cloud computing, Computer and Network Security and Cryptography.

SADQI Yassine received his Ph.D degree in the field of Computer Science and Distributed Systems at Ibn Zoher University in 2012. His research interest is Web Applications Security, Computer Security and Cryptography. He is an assistant professor at Polydisciplinairy Faculty,Sultan Moulay Slimane University, Beni Mellal, Morocco.

Privacy-preserving TPA Auditing Scheme Based on Skip List for Cloud Storage

Haichun Zhao[1,2], Xuanxia Yao[1], and Xuefeng Zheng[1]
(Corresponding author: Xuanxia Yao)
School of Computer and Communication Engineering, University of Science and Technology Beijing[1], Beijing, China
School of Computer Science and Technology, Inner Mongolia University for Nationalities[2], Tongliao, China
(Email: yaoxuanxia@163.com)
(Received Jan. 3, 2018; Revised and Accepted June 15, 2018; First Online Jan. 25, 2019)

Abstract

Recently, researchers have proposed several privacy-preserving public auditing schemes to remotely check the integrity of outsourced data based on homomorphic authenticators, random block sampling and random masking techniques. However, almost all these schemes require users to maintain tables related to the block index. These tables are difficult to maintain, especially when the outsourced data is frequently updated. In this paper, we propose a privacy-preserving public auditing scheme with the support of dynamics using rank-based authenticated skip list for the integrity of the data in cloud storage, of which users do not need to maintain the relevant table. And we give a formal security proof for data integrity guarantee and analysis for privacy-preserving property of the audit protocol. The performance analysis demonstrates that our scheme is highly efficient.

Keywords: Audit Protocol; Cloud Storage; Privacy-preserving; Public Auditing; Rank-based Authenticated Skip List

1 Introduction

Cloud computing has many advantages; this has led to an increasing number of individuals and companies choosing to store their data and conduct their business using cloud-based services [22]. Unlike traditional systems, users lose their physical control over their data. Although the cloud infrastructure is significantly more reliable than personal computing devices, data security/privacy is still one of the core considerations for users when adopting cloud services because of the internal and external threats associated with cloud services [1, 35, 38]. Therefore, researchers have proposed various security models and schemes to overcome the issue of data integrity auditing [3, 12–15, 18, 20, 27, 29–31, 33, 34, 36, 37].

The public auditable schemes allow external parties, in addition to the user, to audit the integrity of outsourced data; however, this could potentially leak the user's data to auditors. Hence, researchers have proposed privacy-preserving public auditing schemes to avoid auditors learning user's data in the auditing phase. The construction of the signatures in some of these schemes involve the block index information i, such as $H(name \parallel i)$ or $H(B_i \parallel V_i \parallel R_i)$ [30, 36, 37]. Users need to maintain a table in the local storage for each file, such as map-version Table [5] or index-hash table [36,37]. The table is also sent to the third-party auditor (*TPA*) before the data is audited. If the table is corrupted, effective audits or dynamic operations cannot be conducted on the outsourced data. In addition, if a large file is stored in cloud storage server (*CSS*) and undergoes frequent insert and delete operations, the block index will continue to increase and become very large. This is because the block index cannot be reused. Consequently, it becomes increasingly difficult for users to maintain the table. To address the problem, the index i is removed, and $H(m_i)$ is used in constructing the signature for block m_i to prevent replay attack on the same hash values. To support privacy-preserving *TPA* auditing, $(H(m_i))^{\alpha/\beta}$ is used in the signature construction and assigned to the data item value for the leaf node of the skip list [9].

In this paper, a secure public auditing algorithm is proposed with the support of dynamics using a rank-based authenticated skip list [9] for the outsourced data. The contributions of this paper can be summarized as follows:

1) A privacy-preserving public auditing scheme which fully supports dynamics by employing rank-based authenticated skip list is proposed. $(H(m_i))^{\alpha/\beta}$ is used as the data item of the bottom node of the skip list to realize privacy-preserving.

2) Based on the cryptography reduction theory [16, 21] and *CPoR's* model [27] a formal security proof

is given for the integrity guarantee of outsourced data and privacy-preserving property of the auditing phase for the scheme.

The remainder of the paper is organized as follows. Section 2 contains the related work. Section 3 introduces the system model and our design goals. In Section 4, we elaborate our proposed scheme. Section 5 analyzes the security and performance of our scheme. The conclusion is given in Section 6.

2 Related Work

Ateniese *et al.* proposed the provable data possession (*PDP*) model, which can be used for remotely checking data integrity [2, 3]. This model can generate probabilistic proofs of possession by randomly sampling data blocks from the server, in which the tags of the sampled blocks can be aggregated into a single value using homomorphic verifiable tags(*HVTs*). It is believed to be the first scheme to provide blockless verification and public verifiability at the same time. Erway *et al.* proposed dynamic PDP (*DPDP*), which applies the structure of *rank-based authenticated skip list* to ensures the integrity of the tags using the skip list structure and the integrity of the blocks by their tags. This scheme effectively supports provable secure updates to the remotely stored data [9]. Juels and Kaliski presented the proof of retrievability (*PoR*) model. This model ensures both the possession and the retrievability of outsourced data by using spot-checking and error-correcting codes. However, the number of audit challenges a user can perform is predetermined and public auditability is not supported in [15].

Shacham *et al.* designed a compact version of *PoR* (*CPoR*) [27] and proved the security of their scheme against arbitrary adversaries in the Juel-Kaliski model. The construction of the publicly verifiable *CPoR* scheme is based on Boneh-Lynn-Shacham (*BLS*) signatures [8]. Wang *et al.* proposed a public auditing scheme that supports dynamic data operations in [31]. The authentication information of the scheme is managed using the Merkle hash tree (MHT) [23], in which the leaf nodes are the values of $H(m_i)$(m_i is the i-th block of the file). To prevent *TPA* extracting data content from the collected information, they designed a privacy-preserving public auditing scheme using a random mask technique to blind the response information in the follow-up work [30]. But its description for the dynamics is ambiguous. Zhu *et al.* proposed another privacy-preserving public auditing scheme which supported dynamic data updates employing an index-hash table [36]. However, in these two privacy-preserving schemes, block index related information is involved in the signature construction. Users are required to maintain a relevant table. To guarantee the integrity of the multiple replicas in cloud, Curtmola *et al.* proposed the replication-based remote data auditing scheme, called Multiple-Replica PDP (*MR-PDP*), which extends the (single-copy) *PDP* scheme for overcoming the collusion attack in a multi-server environment. However, *MR-PDP* only supports private verification [7].

Barsoum *et al.* proposed two multi-copy *DPDP* public auditing schemes, supporting data dynamics based on the MHT and map-version table, respectively. Different copies are generated through encrypting the concatenation of the copy number and file blocks [5]. In the latter, the map-version table must be stored in the local storage of the user and is managed by the user during the various update operations performed on the file. In [34], Yang *et al.* propose a public auditing scheme for shared cloud data in which a group manager is introduced to help members generate authenticators to protect the identity privacy. This method uses two lists to record the members who performed the most-recent modification on each block to achieve the identity traceability. This scheme also achieves data privacy during authenticator generation by employing a blind signature technique. To overcome the issue of resource-constrained users dealing with heavy burdens, Shen *et al.* proposed a cloud storage auditing scheme for group users by introducing a third party medium (*TPM*) to perform time-consuming operations on behalf of users [29]. Utilizing proxy re-signatures and homomorphic linear authenticators, Li *et al.* propose a privacy-preserving cloud data audit scheme that can support key-updating and authenticator-evolving [18].

Researchers have proposed a number of cloud storage auditing schemes in the recent past. All these schemes primarily focus on several different aspects of cloud storage auditing. However, almost none of these schemes address the issue that users need to maintain a block index related table in the local storage for the privacy-preserving public auditing schemes. Users should be "stateless" and must not be required to store and update the table between different dynamic operations, since such table is difficult to maintain.

3 Problem Statements

3.1 System Model

The auditing system for cloud storage involves cloud users, *CSS* and *TPA* as shown in Figure 1. The cloud user is the data owner, who has large amount of data to be stored in the *CSS*. The users can access and dynamically update their data in the *CSS* by interacting with the *CSS*. The *CSS*, which is managed by the cloud service provider (*CSP*), has significant storage space and massive amount of computational resources. The users' data is stored in the cloud storage and is managed and maintained by the *CSP*. The *TPA* has expertise and capabilities that users do not have and can audit the users' outsourced data in the *CSS* on behalf of users at the users' request.

To ensure the integrity and correctness of the users' outsourced data, users need to make periodic checks. To save computation resources and network bandwidth, users can delegate the *TPA* to perform the periodic data integrity verification. However, users do not want informa-

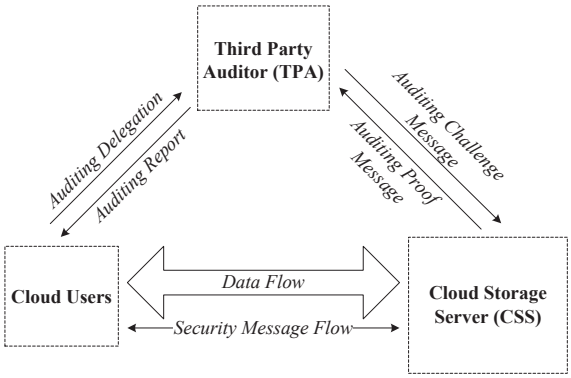


Figure 1: The cloud storage architecture includes the *CSS*, the cloud users and the *TPA*

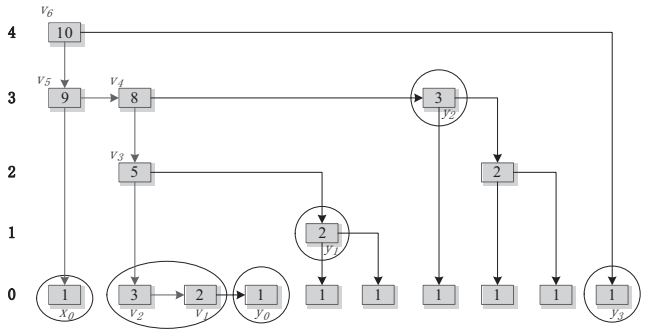


Figure 2: Example of a rank-based skip list

tion from their data to be learned by the *TPA* during the auditing process.

In this model, it is assumed that the cloud server does not have the incentive to reveal their hosted data to any external entity. It is also assumed that the *TPA* has no incentive to collude with either the *CSP* or the user during the auditing process. However, it is interested in the users' data.

3.2 Design Goals

In the aforementioned model, a scheme is proposed in which the design goals can be summarized as follows [19]:

1) Public auditability: To allow any authorized *TPA* to verify the integrity of the cloud data without retrieving a copy of the whole data;
2) Storage correctness: To ensure that there no *CSP* exists that can pass the audit of the *TPA* without storing cloud users' data intact;
3) Privacy preserving: To ensure that it is infeasible for the *TPA* to recover the user's data from the information collected during the auditing phase;
4) High performance: The *TPA* can perform data auditing with minimum communication and computation overhead;
5) Dynamic data: To allow the data owners to modify, insert and delete data blocks in the cloud storage when they want to update their data at any time;
6) Batch auditing: The *TPA* can audit the data of different users at the same time.

4 The Proposed Construction

4.1 Preliminaries

Relevant functions. A pseudo-random function (*PRF*) f and a pseudo-random permutation (*PRP*) π are used with the following parameters [3]:

$$f_k : \{0,1\}^{log_2 n} \times K \rightarrow \{0,1\}^l;$$
$$\pi_k : \{0,1\}^{log_2 n} \times K \rightarrow \{0,1\}^{log_2 n}.$$

Bilinear maps. Suppose a group G is a Gap Diffie-Hellman (*GDH*) group with prime order p. G_T is another multiplicative cyclic group with prime order p. Then, the bilinear map is a map $e : G \times G \rightarrow G_T$ with the following properties [8]:

1) Bilinearity – $\forall u, v \in G, a, b \in Z_p, e(u^a, v^b) = e(u, v)^{ab}$;
2) Non-degeneracy – $e(g, g) \neq 1$, where g is a generator of G;
3) Computability – e should be efficiently computable.

The following scheme description uses the symmetric bilinear map for the purpose of simplicity. The asymmetric bilinear map is in the form of $e : G_1 \times G_2 \rightarrow G_T$.

Rank-Based Skip List [9,11,24]. The main information related to i-th node v on level 0 (bottom-level) includes: the level of i-th node $l(v)$, the rank of i-th node $r(v)$, the data item of i-th node $T(m_i)$ and the label of i-th node $f(v)$; that on non-bottom level includes: the level of the node $l(v)$, the rank of the node $r(v)$, the label of the node $f(v)$; In addition to these, each node contains some information related to the structure of the skip list, such as, right and down pointers.

The rank value of a node indicates the number of the reachable bottom nodes (or leaf nodes) departing from the node. The rank of a Null node equals 0. The location of each bottom node can be calculated from the rank values of the relevant nodes.

The label value of a node on bottom-level is

$$f(v) = h_2(l(v) \parallel r(v) \parallel T(m_i) \parallel f(right(v)))$$

and that on non-bottom level is

$$f(v) = h_2(l(v) \parallel r(v) \parallel f(down(v)) \parallel f(right(v)))$$

where the symbol "||" denotes concatenation, $f(down(v))$ and $f(right(v))$ are the label of the down and right node

of v, respectively. The label value of a Null node is 0. The function $h_2(\cdot)$ is a collision-resistant cryptographic hash function. Users hold the label $f(s)$ of the top leftmost node (or start node) of the skip list. The $f(s)$ is called the basis (or *root*). It is equivalent to the user's verification metadata.

To obtain the proof information of some block i, the skip list needs traversing from the start node v_k to the node v_1 associated with block i through the rank of the nodes. The reverse path $v_1, \cdots, v_k$ is called *verification path* of the block i, as shown in Figure 2. The information of the nodes $x_0, y_0 - y_3$ and $v_1 - v_2$ is used as auxiliary authentication information (AAI) for calculating each rank and label value from v_1 to v_k on the *verification path*.

The proof of a block is composed of a sequence of tuples made of the relevant information of each node on the *verification path*. That is, the proof for block i with data $T(m_i)$ is a sequence $\Pi_i = (A(v_1), \cdots, A(v_k))$ where $A(v_j) = (l(v_j), q(v_j), d(v_j), g(v_j)), 1 \le j \le k$, from which we can get the AAI. The $l(v_j)$ is the level of the node and Boolean $d(v_j)$ indicates whether v_{j-1} is to the right or below v_j. The value of $g(v_j)$ is used to calculate the label of the corresponding node along the *verification path*. For the non-bottom level nodes, if $d(v_j) = rgt$, then $g(v_j) = f(dwn(v_j))$, else if $d(v_j) = dwn$, then $g(v_j) = f(rgt(v_j))$. For bottom-level nodes $v_j (j > 1)$ on the *verification path*, the value of $g(v_j)$ is the data item of the node. The value of $g(v_1)$ is the label of the right node of v_1. For nodes at the bottom-level, $q(v_1)$ is the sum of the rank of the right node of v_1 and 1, this 1 means that the node v_1 itself is also a reachable node on the bottom-level. The value of $q(v_j)$ of each node on the left side of the node v_1 at bottom-level is 1. For non-bottom level nodes, if the node v_{j-1} is the right (or down) node of v_j, then $q(v_j) = r(dwn(v_j))$ (or $q(v_j) = r(rgt(v_j))$).

4.2 The Privacy-preserving Scheme

The notions proposed in [3, 15, 27, 28, 30, 31, 36] were followed in this study. The proposed scheme is based on *CPoR's* model [27] and the relevant method in [25].

The scheme consists of two algorithms $KeyGen(1^k), St(sk, F)$ and an interactive audit protocol $Audit(CSP, TPA)$.

Let $S = (p, G, G_T, e)$ be a bilinear map group system with randomly selected generators $g, h \in_R G$, where G, G_T are two groups of large prime order p. $H(\cdot)$ is a secure map-to-point hash function: $\{0,1\}^* \to G$, which maps strings uniformly to G. Another hash function $h_1(\cdot) : G_T \to Z_p$ maps the group element of G_T uniformly to Z_p.

$KeyGen(1^k)$: This randomized algorithm generates the public and secret parameters. The cloud user chooses a random signing key pair (*spk,ssk*) and two random $\alpha, \beta \in_R Z_p$. The secret parameter is *sk*= (α, β, ssk) and the public parameter is *pk*=(*g*,*h*,*X*,*Y*) , where $X = h^\alpha, Y = h^\beta \in G$.

$St(sk, F)$: The data file F is split into $n \times s$ sectors $F = \{m_{ij}\}^{n \times s}, m_i = \{m_{ij}\}_{1 \le j \le s}, m_{ij} \in Z_p$. The cloud user chooses s random $\tau_1, \cdots, \tau_s \in Z_p$ as the secret of the file and computes $u_j = g^{\tau_j} \in G, 1 \le j \le s$ and authenticator

$$\sigma_i \leftarrow (H(m_i))^\alpha \cdot (\prod_{j=1}^{s} u_j^{m_{ij}})^\beta = ((H(m_i))^{\alpha/\beta} \cdot g^{\sum_{j=1}^{s} \tau_j \cdot m_{ij}})^\beta \quad (1)$$

for each block i. The cloud user constructs a *rank-based authenticated skip list* of which the data item of the i-th bottom node is $(H(m_i))^{\alpha/\beta}, 1 \le i \le n$. Let $\Phi = (\sigma_1, \cdots, \sigma_n)$ and t_0 *be* "$fn \parallel n \parallel u_1 \parallel \cdots \parallel u_s \parallel M_c$", *fn* is chosen by the user uniformly at random from Z_p as the identifier of file F, M_c is the root of the skip list. The cloud user computes $t = t_0 \parallel SSig_{ssk}(t_0)$ as the file tag for F under the private key *ssk*. The user then sends $\{F, \Phi, t\}$ and the skip list to the cloud server and deletes $\{F, \Phi\}$ and the skip list from his local storage. Then the user holds t as the *metadata*.

$Audit(CSP, TPA)$: This is a 3–move protocol between *TPA* and *CSP* as the following:

- *Commitment*(*CSP* $\to$ *TPA*): The *CSP* chooses s random $\lambda_j \in_R Z_p, (1 \le j \le s)$, then computes $T_j = u_j^{\lambda_j}, (1 \le j \le s)$ and sends its commitment, $\{T_j\}_{j \in [1,s]}$, to *TPA*.
- *Challenge*(*TPA* $\to$ *CSP*): The authorized *TPA* first retrieves the file tag t. The *TPA* checks the validity of t via *spk*, and quits by outputting *reject* if the verification fails. Otherwise, the *TPA* recovers the values in t_0. Then *TPA* generates a set of challenge information $Chal = \{c, k_1, k_2\}$ [3], in which c is the number of the data blocks to be audited and k_1, k_2 are randomly chosen keys for the pseudo-random permutation π_k and pseudo-random function f_k, respectively. The π_k and f_k are used to generate c indices $s_j (1 \le j \le c, 1 \le s_j \le n)$ and c relevant coefficients $v_i (i \in \{s_j | 1 \le j \le c\}, v_i \in_R Z_p)$ of the challenged data blocks. Let I denotes the set of c random indices s_j. Let Q be the set $\{(i, v_i)\}_{i \in I}$ of the index and coefficient pairs. Then *TPA* sends *Chal* to the prover *CSP*.
- *Response*(*TPA* $\leftarrow$ *CSP*): Upon receiving the challenge *Chal*, *CSP* chooses a random $r \in_R Z_p$ and calculates

$$\psi = e(g^r, h), \gamma = h_1(\psi), s_j = \pi_{k_1}(j),$$

and

$$v_i = f_{k_2}(j),$$

where

$$1 \le j \le c, i \in \{s_j | 1 \le s_j \le n\}, v_i \in Z_p.$$

Then the *CSP* computes

$$\begin{cases} \sigma \leftarrow \prod_{(i,v_i)\in Q} \sigma_i^{\gamma \cdot v_i} \cdot g^r \\ \mu_j \leftarrow \lambda_j^{-1} \cdot (\gamma \cdot \sum_{(i,v_i)\in Q} v_i \cdot m_{ij} + 1) \end{cases} \tag{2}$$

and sends the response $\theta = (\sigma, \{\mu_j\}_{j\in[1,s]}, \psi)$, the set $\{\Pi_i\}_{i\in I}$ of the proof for every block i and $\{(H(m_i))^{\alpha/\beta)}\}_{i\in I}$ to the *TPA*.

Verification: *TPA* calculates the root R_t from $\{(H(m_i))^{\alpha/\beta}, \Pi_i\}_{i\in I}$ and checks $R_t \stackrel{?}{=} M_c$. If it is not true, *TPA* outputs *reject*, otherwise *TPA* can check

$$e(\sigma,\ h) \stackrel{?}{=} \psi \cdot e(\prod_{(i,v_i)\in Q}((H(m_i))^{\alpha/\beta})^{v_i\cdot\gamma} \cdot \prod_{j=1}^{s} T_j^{\mu_j} \cdot u_j^{-1},\ Y) \tag{3}$$

If it holds, *TPA* outputs *accept*, otherwise *reject*.

4.3 Support for Dynamic Data Operation

Merkle hash tree can perfectly work for the static case and also do well when the elements are inserted in a random order for the dynamic case. When it undergoes a sequence of inserting operations in a certain order, the structure of the binary tree may degenerate and the performance may become poor. In this case, the binary tree will need rebalancing continuously with the operations [4,17]. While the skip lists are balanced probabilistically, in dealing with a variety of dynamic operations, the performance of the skip list is relatively stable [26]. So, we choose the *rank-based authenticated skip list* [9] as the authenticated search data structure of the dynamic case [10]. Through this structure, various dynamic operations can be efficiently performed, the order of data blocks in the file can be guaranteed not to be changed, the integrity of $(H(m_i))^{\alpha/\beta}$ can be ensured and then the integrity of the signatures and the data blocks can be ensured.

Now we describe the dynamic data operations. Our scheme can fully handle block-level dynamic operations including modification ($'M'$), insertion ($'I'$) and deletion ($'D'$) for the outsourced data. Each operation affects only nodes along a verification path in the skip list. We assume that the file F, the signatures of data blocks Φ and the corresponding skip list with the elements $(H(m_i))^{\rho}(1 \le i \le n, \rho = \alpha/\beta)$ have been stored in the cloud server. The user keeps the root as *verification metadata*, which is the label of the start node of the skip list.

Data modification: We assume that the user wants to modify the i-th data block m_i to $m_i^{'}$. Firstly, the user sends a query "$Prepareupdate = (i)$" to the server to get the message which includes $H(m_i)$ and the proof Π_i of block i. After receiving these information, the user computes $(H(m_i))^{\rho}$ and generates root S. Then the user checks $M_c \stackrel{?}{=} S$. If it is not true, output *reject*, otherwise the user computes the new block signature $\sigma_i \leftarrow ((H(m_i^{'}))^{\alpha/\beta} \cdot \prod_{j=1}^{s} u_j^{m_{ij}^{'}})^{\beta}$. Then, he constructs an update request message "$Update = ('M', i, m_i^{'}, \sigma_i^{'}, H(m_i^{'})^{\rho})$" and sends it to the server. Upon receiving the request, the server runs $PerformUpdate(F, \Phi, Update)$. Through the procedure the server completes the following tasks:

1) Replaces m_i and σ_i with $m_i^{'}$ *and* $\sigma_i^{'}$, respectively;
2) Replaces $(H(m_i))^{\rho}$ with $(H(m_i^{'}))^{\rho}$ of the leaf node i, then updates the labels of the affected nodes and generates the new root $M_c^{'}$.

Finally the server returns $M_c^{'}$ to the user. Then the user generates the new root $S^{'}$ using $\Pi_i, (H(m_i^{'}))^{\rho}$ and compares it with $M_c^{'}$ to check whether the server has performed the modification operation as required or not. If it is not true, output *reject*, otherwise output *accept*. Then, the user replaces M_c with $M_c^{'}$ as the new root metadata and deletes *Update* and $m_i^{'}$ from its local storage.

Data insertion: Data insertion means inserting a new block after some specified position in the file F. Suppose the user wants to insert a block $m_{i+1}^{'}$ after the i–th block m_i. Firstly, the user sends a query "$Prepareupdate = (i)$" to the server, then the server returns $H(m_i)$ and the proof Π_i of block i. Next, the user computes $(H(m_i))^{\rho}$ and generates root S using $\{\Pi_i, H(m_i)^{\rho}\}$. Then the user checks $M_c \stackrel{?}{=} S$. If it is not true, output *reject*, otherwise the user computes the new block signature $\sigma_{i+1}^{'} = ((H(m_{i+1}^{'}))^{\alpha/\beta} \prod_{j=1}^{s} u_j^{m_{i+1,j}^{'}})^{\beta}$ and determines the height of the tower of the skip list associated with the new block. Finally he constructs an update request message "$Update = ('I', l, i, m_{i+1}^{'}, \sigma^{'}, H(m_{i+1}^{'})^{\rho})$" and sends it to the server, where 'l' denotes the height of the tower related to the new node. Upon receiving the request, the server runs $PerformUpdate(F, \Phi, Update)$. The server completes the following tasks:

1) The server stores data block $m_{i+1}^{'}$ and its signature $\sigma_{i+1}^{'}$;
2) The server adds a leaf node after the position i of which the data item is $(H(m_{i+1}^{'}))^{\rho}$ according to the height l, then updates the labels, levels and ranks of the affected nodes and generates the new root $M_c^{'}$ based on the updated skip list.

The server sends to the user $M_c^{'}$ in response. Then the user generates the new root $S^{'}$ using $\{\Pi_i, (H(m_{i+1}^{'}))^{\rho}\}$ and compares it with $M_c^{'}$ to check whether the server has performed the insertion operation as required or not. If it is not true, output *reject*, otherwise output *accept*. Then, the user replaces M_c with $M_c^{'}$ as the new root

metadata and deletes *Update* and m'_{i+1} from its local storage.

Data deletion: Data deletion refers to deleting a specified data block from the file. The corresponding element in the skip list will be deleted at the same time. Data deletion is the opposite operation of data insertion. However, the parameters specified by the user don't include the tower height. The details of the operation procedure are similar to that of data modification and insertion, so we omit them here.

4.4 Support for Batch Auditing

When the *TPA* simultaneously copes with different auditing delegations from different D users on different D files respectively, we can extend our scheme to implement batch auditing tasks. If the i in the Q is within the range of the number of blocks of the file, the auditing for the file can be added into the batch auditing. The batch auditing scheme can reduce the number of relatively expensive pairing operations from *2D* to *D+1*.

The k^{th} user randomly chooses parameters $u_{k,j} \in_R G, 1 \le k \le D, 1 \le j \le s$. His/her secret key and corresponding public key are denoted as $sk_k = (\alpha_k, \beta_k, ssk_k)$ and $pk_k = (X_k, Y_k, spk_k)$. The user's outsourced file is $F_k = \{m_{k,i,j}\}, (1 \le k \le D, 1 \le i \le n, 1 \le j \le s)$, the file name is fn_k and the tag of the file is $t_k = t_{k,0} \parallel SSig_{ssk_k}(t_{k,0})$. The signature of the block i is $\sigma_{k,i} = ((H(m_{k,i}))^{\alpha_k/\beta_k} \cdot \prod_{j=1}^{s} u_{k,j}^{m_{k,i,j}})^{\beta_k}$. The root of the corresponding skip list is $M_{k,c}$.

The *CSP* chooses $\lambda_{k,j} \in_R Z_p$, then computes $T_{k,j} = u_{k,j}^{\lambda_{k,j}}$ as the commitments for each user. The *TPA* chooses the challenge $Chal = \{c, k_1, k_2\}$ and sends *Chal* to *CSP*. After receiving *Chal*, the *CSP* gets $Q = \{(i, v_i)\}_{i \in I}$, chooses randomly $r_k \in_R Z_p$ and calculates $\psi_k = e(g^{r_k},\ h), \gamma_k = h_1(\psi_k)$ and

$$\begin{cases} \sigma_k \leftarrow \prod_{(i,v_i)\in Q} \sigma_{k,i}^{\gamma_k \cdot v_i} \cdot g^{r_k} \\ \mu_{k,j} \leftarrow \lambda_{k,j}^{-1} \cdot (\gamma_k \cdot \sum_{(i,v_i)\in Q} v_i \cdot m_{k,i,j} + 1) \end{cases} \tag{4}$$

The *CSP* sends $\theta_k = (\{\sigma_k\}_{1\le k\le D}, \{\mu_{k,j}\}_{1\le k\le D, 1\le j\le s}, \{\psi_k\}_{1\le k\le D})$, the set $\{\Pi_{k,i}\}_{1\le k\le D, i\in I}$ of the proof for block $m_{k,i}$ and $\{(H(m_{k,i}))^{\alpha_k/\beta_k}\}_{1\le k\le D, i\in I}$ to the *TPA*.

After receiving the response from the *CSP*, the *TPA* calculates the root $R_{k,t}$ from $\{(H(m_{k,i}))^{\alpha_k/\beta_k}, \Pi_{k,i}\}_{1\le k\le D, i\in I}$ and checks $R_{k,t} \stackrel{?}{=} M_{k,c}$ for every file. If it is not true, *TPA* outputs *reject*, otherwise *TPA* can check

$$e(\prod_{k=1}^{D} \sigma_k,\ h) \stackrel{?}{=} \prod_{k=1}^{D} (\psi_k \cdot e(\prod_{(i,v_i)\in Q} ((H(m_{k,i}))^{\alpha_k/\beta_k})^{v_i\cdot\gamma_k} \cdot \prod_{j=1}^{s} T_{k,j}^{\mu_{k,j}} \cdot u_{k,j}^{-1},\ Y_k)) \tag{5}$$

If it holds, *TPA* outputs *accept*, otherwise *reject*.

5 Evaluation

5.1 Security Evaluation

Completeness property: For each random challenge Q and its corresponding correct responses, the completeness of the protocol can be elaborated as follows:

$$e(\sigma, h) \stackrel{?}{=} \psi \cdot e(\prod_{(i,v_i)\in Q} ((H(m_i))^{\alpha/\beta})^{v_i\cdot\gamma} \cdot \prod_{j=1}^{s} T_j^{\mu_j} \cdot u_j^{-1},\ Y)$$

The right side

$$\begin{aligned} &= e(\prod_{(i,v_i)\in Q} (H(m_i))^{(\alpha/\beta)\cdot v_i\cdot\gamma} \\ &\quad \cdot \prod_{j=1}^{s} u_j^{\gamma\cdot\sum_{(i,v_i)\in Q} v_i\cdot m_{ij}+1} \cdot u_j^{-1},\ Y) \cdot \psi \\ &= e(\prod_{(i,v_i)\in Q} (H(m_i))^{(\alpha/\beta)\cdot v_i\cdot\gamma} \\ &\quad \cdot \prod_{j=1}^{s} u_j^{\gamma\cdot\sum_{(i,v_i)\in Q} v_i\cdot m_{ij}},\ Y) \cdot \psi \\ &= e(\prod_{(i,v_i)\in Q} (H(m_i))^{(\alpha/\beta)\cdot v_i\cdot\gamma} \\ &\quad \cdot \prod_{(i,v_i)\in Q} (\prod_{j=1}^{s} u_j^{m_{ij}})^{v_i\cdot\gamma},\ Y) \cdot \psi \\ &= e(\prod_{(i,v_i)\in Q} ((H(m_i))^{\alpha/\beta} \cdot \prod_{j=1}^{s} u_j^{m_{ij}})^{v_i\cdot\gamma\cdot\beta} \cdot g^r,\ h) \end{aligned}$$

= The left side of the equation

So the equation means that the protocol is valid for the correct responses.

Soundness property: The soundness property means that a false response will not be accepted as the correct. In this context, it means that the *CSS* cannot generate a valid response to the *TPA's* challenge if the outsourced data is not stored well.

Theorem 1. *If the CSS passes the verification of the Audit protocol, it must indeed store the specified data intact.*

Following from the proof of *CPoR* [[27], Theorem 4.2], we give a proof of Theorem 1 in the random oracle model.

Proof. To prevent the *TPA* from extracting the value of σ_i from $\prod_{(i,v_i)\in Q} \sigma_i^{v_i\cdot\gamma}$, we blind it with g^r at each instance. To prove that the cloud server cannot falsify $\sigma, \{\mu_j\}_{1\le j\le s}$, we assume that the response information contains g^r instead of ψ and also contains $\lambda_j, (1 \le j \le s)$, corresponding to the commitment. □

There are a challenger and an adversary , and the latter is a malicious *CSP*. The challenger constructs a simulator S that will simulate the entire environment of the scheme for the adversary A. For any file F on which it previously made *St* query, the adversary A

can perform the *Audit* protocol with the challenger. In these executions of the protocol, the simulator S plays the part of the verifier and the adversary A plays the part of the prover: $S(pk, sk, t) \rightleftharpoons A$.

For some file F, if the adversary A can successfully forge the aggregate signature $\sigma^{'}$ with a non-negligible probability resulting in $\sigma^{'} \neq \prod_{(i,v_i)\in Q} \sigma_i^{v_i\cdot\gamma} \cdot g^r$ and successfully pass the verification, the simulator can make use of the adversary to solve the Computational Diffie-Hellman problem.

The simulator is given as input values $h, X = h^{\alpha}, Y = h^{\beta}$, and its goal is to output $h^{\alpha\cdot\beta}$.

Let $H : \{0,1\}^* \rightarrow G$ be a hash function which will be modeled as a random oracle. The simulator programs the random oracle H. When answering queries from the adversary, it chooses a random $\varphi \xleftarrow{R} Z_p$ and respond with $h^{\varphi} \in G$. When answering the queries of the form $H(m_i)$, the simulator programs it in a special way described below.

For each $j, 1 \leq j \leq s$, the simulator chooses random values $\eta_j, \theta_j \xleftarrow{R} Z_p$ and sets $u_j \leftarrow X^{\eta_j} \cdot h^{\theta_j}$.

For each $i, 1 \leq i \leq n$, the simulator chooses a random value $r_i \xleftarrow{R} Z_p$, and programs the random oracle at i as

$$H(m_i) = h^{r_i} / Y^{\sum_{j=1}^{s} \eta_j \cdot m_{ij}}.$$

Now the simulator computes:

$$\begin{aligned}
\sigma_i &= (H(m_i))^{\alpha} \cdot (\prod_{j=1}^{s} u_j^{m_{ij}})^{\beta} \\
&= (h^{r_i}/(Y^{\sum_{j=1}^{s} \eta_j \cdot m_{ij}}))^{\alpha} \cdot (\prod_{j=1}^{s} (X^{\eta_j} \cdot h^{\theta_j})^{m_{ij}})^{\beta} \\
&= (h^{r_i}/(Y^{\sum_{j=1}^{s} \eta_j \cdot m_{ij}}))^{\alpha} \\
&\qquad \cdot (X^{\sum_{j=1}^{s} \eta_j \cdot m_{ij}} \cdot h^{\sum_{j=1}^{s} \theta_j \cdot m_{ij}})^{\beta} \\
&= h^{\alpha\cdot r_i} \cdot h^{\beta \cdot \sum_{j=1}^{s} \theta_j \cdot m_{ij}} \\
&= X^{r_i} \cdot Y^{\sum_{j=1}^{s} \theta_j \cdot m_{ij}} \qquad (6)
\end{aligned}$$

The challenger keeps a list of its responses to *St* queries made by the adversary. Now the challenger observes each instance of the *Audit* protocol with the adversary A. If in any of these instances the adversary is successful (*i.e.*, the verification equation holds), but the adversary's aggregate signature $\sigma^{'} \neq \prod_{(i,v_i)\in Q} \sigma_i^{v_i\cdot\gamma} \cdot g^r$, the challenger declares failure and aborts.

Suppose $Q = \{(i, v_i)\}_{i\in I}$ is the query that causes the challenger to abort, and the adversary's response to that query is $\mu_1^{'}, \cdots, \mu_s^{'}$ together with $\sigma^{'}$. Let the expected response be $\mu_1, \cdots, \mu_s$ and σ. By the correctness of the scheme, the expected response satisfies the verification equation, *i.e.*, that

$$\begin{aligned}
&e(\sigma,\ h)/\psi \\
=&e(\prod\nolimits_{(i,v_i)\in Q}((H(m_i))^{\alpha/\beta})^{v_i\cdot\gamma} \cdot \prod\nolimits_{j=1}^{s} T_j^{\mu_j} \cdot u_j^{-1},\ Y) \qquad (7)
\end{aligned}$$

Because the challenger aborts, we know that $\sigma \neq \sigma^{'}$ and that $\sigma^{'}$ passes the verication equation, *i.e.* that

$$\begin{aligned}
&e(\sigma^{'},\ h)/\psi \\
=&e(\prod\nolimits_{(i,v_i)\in Q}((H(m_i))^{\alpha/\beta})^{v_i\cdot\gamma} \cdot \prod\nolimits_{j=1}^{s} T_j^{\mu_j^{'}} \cdot u_j^{-1},\ Y) \qquad (8)
\end{aligned}$$

Observe that if $\mu_j^{'} = \mu_j$ for each j, we can get $\sigma^{'} = \sigma$, which contradicts our assumption above. Therefore, if we define $\Delta\mu_j \stackrel{\text{def}}{=} \mu_j^{'} - \mu_j$ for $1 \leq j \leq s$, it must be the case that at least one of $\{\Delta\mu_j\}$ is nonzero. Let $\sigma^* = \prod_{(i,v_i)\in Q} \sigma_i^{v_i}$ and $\mu_j^* = \sum_{(i,v_i)\in Q} v_i \cdot m_{ij}$. So, dividing the Equation (8) by the Equation (7) , we obtain

$$\begin{aligned}
&e((\sigma^{*^{'}})^{\gamma}/(\sigma^*)^{\gamma},\ h) \\
=&e(\prod\nolimits_{j=1}^{s} u_j^{\gamma\cdot\Delta\mu_j^*},\ Y) \\
=&e(\prod\nolimits_{j=1}^{s} (X^{\eta_j} \cdot h^{\theta_j})^{\gamma\cdot\Delta\mu_j^*},\ Y) \\
=&e(\prod\nolimits_{j=1}^{s} X^{\gamma\cdot\eta_j\cdot\Delta\mu_j^*},\ Y) \cdot e(\prod\nolimits_{j=1}^{s} h^{\gamma\cdot\theta_j\cdot\Delta\mu_j^*},\ Y) \\
=&e(X^{(\sum_{j=1}^{s}\eta_j\cdot\Delta\mu_j^*)\cdot\gamma},\ Y) \cdot e(h^{(\sum_{j=1}^{s}\theta_j\cdot\Delta\mu_j^*)\cdot\gamma},\ Y) \qquad (9)
\end{aligned}$$

$$\begin{aligned}
&e((\sigma^{*^{'}})^{\gamma} \cdot ((\sigma^*)^{\gamma})^{-1} \cdot Y^{-\gamma\cdot(\sum_{j=1}^{s}\theta_j\cdot\Delta\mu_j^*)},\ h) \\
=&e((X^{\gamma\cdot(\sum_{j=1}^{s}\eta_j\cdot\Delta\mu_j^*)})^{\beta},\ h) \qquad (10)
\end{aligned}$$

So, if $\sum_{j=1}^{s} \eta_j \cdot \Delta\mu_j^* \neq 0$, we see that we have found the solution to the computational Diffie-Hellman problem:

$$h^{\alpha\cdot\beta} = ((\sigma^{*^{'}})^{\gamma} \cdot (\sigma^{*\gamma})^{-1} \cdot Y^{-\gamma\cdot(\sum_{j=1}^{s}\theta_j\cdot\Delta\mu_j^*)})^{1/(\gamma\cdot\sum_{j=1}^{s}\eta_j\cdot\Delta\mu_j^*)}$$

Except the case that $\sum_{j=1}^{s} \eta_j \cdot \Delta\mu_j^*$ is equal to zero. However, we have already realized that not all of $\{\Delta\mu_j^*\}$ can be zero, and the values of $\{\eta_j\}$ are information that is theoretically hidden from the adversary, so $\sum_{j=1}^{s} \eta_j \cdot \Delta\mu_j = 0$ is only with the probability $1/p$, which is negligible.

As demonstrated before, we know $\sigma^{'} = \sigma$. Equating the verifications gives us

$$e(\sigma,\ h) = e(\sigma^{'},\ h),$$

from which using μ and $\mu^{'}$ we get that

$$\begin{aligned}
e(\prod\nolimits_{j=1}^{s} u_j^{\gamma\cdot\mu_j{}^*}),\ Y) &= e(\prod\nolimits_{j=1}^{s} u_j^{\gamma\cdot\mu_j^{*^{'}}}, Y) \\
\prod_{j=1}^{s} u_j^{\gamma\cdot\Delta\mu_j{}^*} &= 1 \\
\prod\nolimits_{j=1}^{s} (X^{\eta_j} \cdot h^{\theta_j})^{\gamma\cdot\Delta\mu_j{}^*} &= 1 \\
X^{\sum_{j=1}^{s}\gamma\cdot\eta_j\cdot\Delta\mu_j{}^*} \cdot h^{\sum_{j=1}^{s}\gamma\cdot\theta_j\cdot\Delta\mu_j{}^*} &= 1 \\
X^{\sum_{j=1}^{s}\gamma\cdot\eta_j\cdot\Delta\mu_j{}^*} &= h^{-\sum_{j=1}^{s}\gamma\cdot\theta_j\cdot\Delta\mu_j{}^*}
\end{aligned}$$

So we find the solution to the discrete logarithm problem,

$$\alpha = -(\sum_{j=1}^{s} \gamma \cdot \theta_j \cdot \Delta\mu_j^*)/(\sum_{j=1}^{s} \gamma \cdot \eta_j \cdot \Delta\mu_j^*),$$

except the case that $\sum_{j=1}^{s} \eta_j \cdot \Delta\mu_j^*$ is equal to zero. While not all of $\{\Delta\mu_j^*\}$ can be zero, and the values of $\{\eta_j\}$ are information that is theoretically hidden from the adversary, so $\sum_{j=1}^{s} \eta_j \cdot \Delta\mu_j^* = 0$ is only with probability $1/p$, which is negligible. This completes the proof of the Theorem 1.

Privacy-Preserving Property: The privacy-preserving property means that TPA cannot extract users' data from the information gleaned during the auditing phase.

Theorem 2. *The TPA cannot extract users' data from the CSP's response* θ and $\{(H(m_i))^{\alpha/\beta}\}_{i\in I}$.

Proof. The m_i, α and β are all hidden from the *TPA*, so $H(m_i)$ cannot be determined from $(H(m_i))^{\alpha/\beta}$. Although $e(H(m_i), X)$ is equal to $e((H(m_i))^{\alpha/\beta}, Y)$, $H(m_i)$ cannot be calculated from it either. Because the isomorphism $f_Q : G \to G_T$ by $f_Q(P) = e(P, Q)$ is believed to be one-way function [6], when given $f_Q(P)$, it is infeasible to find its inverse. In addition, X can be removed from the pk in a concrete implementation. Therefore, it is hard to recover m_i from $(H(m_i))^{\alpha/\beta}$. Similarly, it is hard to extract σ_i from σ.

Every λ_j is randomly chosen by *CSP*, the λ_j^{-1} is the inverse element of it. Both of them are hidden from *TPA*. The $\sum_{i,v_i\in Q} v_i m_{ij}$ is blinded with λ_j^{-1}, so μ_j is uniformly distributed in Z_p for every response. Although *TPA* can obtain enough linear combinations of the data block m_i and its coefficient v_i, he must firstly obtain λ_j^{-1} if he wants to get μ_j^*. The λ_j can be calculated from $T_j = u_j^{\lambda_j}, (1 \le j \le s)$. But this means to solve the discrete logarithm problem (*DLP*). Due to the hardness assumption of *DLP*, *TPA* cannot get λ_j. So it is hard to obtain users' data from $\mu_j, (1 \le j \le s)$. This completes the proof of the Theorem 2. □

5.2 Performance Analysis

In order to elaborate the computation overhead of each entity, we specify some notations for the basic computational operations in the Table 1 [32].

We compared the two typical privacy-preserving data auditing schemes with that of ours in Table 2 for the computational cost of the user, *CSP*, and *TPA*, respectively. Here, n, s, and c are the number of data blocks, number of sectors and number of sampled data blocks, respectively. For the storage and communication overhead of our scheme, we present the following complexity analysis:

Table 1: Notations of relevant operations

Notation	Meaning
$Mult_G^x$	x multiplications in group G
$Mult_{Z_p}^x$	x multiplications in group Z_p
$Hash_{Z_p}^x$	x hash values into group Z_p
$Hash_G^x$	x hash values into group G
$Hash_{D_g}^x$	x times hash function $h_2(\cdot)$, generating message digest
$Add_{Z_p}^x$	x additions in group Z_p
Exp_G^x	x exponentiations g^t, for $g \in G, t \in Z_p$
$Exp_{G_T}^x$	x exponentiations g^t, for $g \in G_T, t \in Z_p$
$Pair_{G_T}^x$	x pairings $e(u, v)$, where $u, v \in G, e(u, v) \in G_T$
PRP_S^x	x pseudo–random permutations in $S = \{0,1\}^{\log_2 n}$
$PRF_{Z_p}^x$	x pseudo–random functions in Z_p

1) The user storage complexity is $O(1)$ and the server storage complexity is $O(n)$.

2) The communication complexity of the challenge phase is $O(1)$ and that of the response phase is $O(\log n)$.

We compared the complexities of the storage and communication of the audit protocol of our scheme with that of two other privacy-preserving schemes in Table 3. The communication complexity in the phase of auditing is $O(\log n)$ in our scheme; however, we could save the maintenance of a table with $O(n)$ complexity of storage space on the user side.

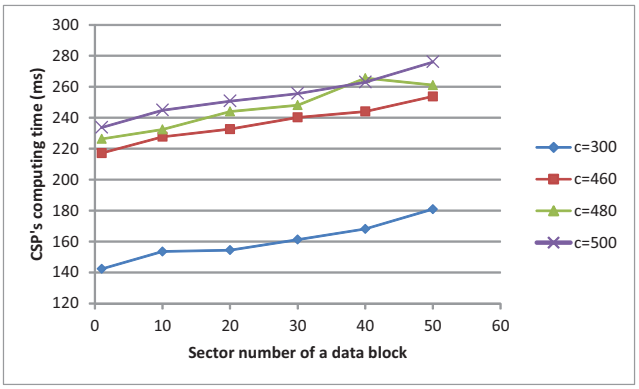


Figure 3: Comparison of computing time for CSP under different s and c

Based on the Pairing-Based Cryptography (PBC) library version 0.5.14, we implement our experiment using C language on an Ubuntu Linux system with an Intel Core i7-4790 CPU running at 3.60GHz with 8GB of RAM and a 7,200 RPM Seagate 1 TB drive. The elliptic curve we choose in the experiment is an MNT curve, with base field size of 159 bits and the embedding degree 6. The length

Table 2: Comparison of computation overhead of different privacy-preserving schemes

	The Computation overhead		
Scheme	**User**	**Server**	**Verifier**
[30]	$Exp_G^{n\cdot(s+2)} + Mult_G^{(n\cdot s)} + Hash_G^n$	$Pair_{G_T}^s + Exp_{G_T}^s + Exp_G^c + Mult_G^{c-1} + Mult_{Z_p}^{(c+1)\cdot s} + Add_{Z_p}^{c\cdot s} + Hash_{Z_p}^1$	$Pair_{G_T}^2 + Exp_G^{s+c+2} + Mult_G^{c+s-1} + Mult_{G_T}^s + Hash_G^c + Hash_{Z_p}^1$
[37]	$Exp_G^{2n+2+s} + Mult_G^n + Mult_{Z_p}^{n\cdot(s+1)} + Add_{Z_p}^{n\cdot(s-1)} + Hash_G^n$	$Pair_{G_T}^1 + Exp_G^{c+s+2} + Mult_G^{c+s-2} + Mult_{Z_p}^{(c+1)\cdot s} + Add_{Z_p}^{c\cdot s}$	$Pair_{G_T}^3 + Exp_G^{c+s} + Mult_G^{c+s-2} + Mult_{G_T}^2 + Hash_G^c$
Our scheme	$Exp_G^{3\cdot n+2} + Mult_G^n + Mult_{Z_p}^{n\cdot s} + Add_{Z_p}^{n\cdot(s-1)} + Hash_G^n$	$Pair_{G_T}^1 + Exp_G^{c+s+2} + Mult_G^c + Mult_{Z_p}^{c+2} + Add_{Z_p}^c + Hash_{Z_p}^1 + PRP_S^c + PRF_{Z_p}^c$	$Pair_{G_T}^2 + Exp_G^{c+s+2} + Mult_G^{c+s} + Hash_{Dg}^{c\cdot(\log n-1)} + PRP_S^c + PRF_{Z_p}^c$

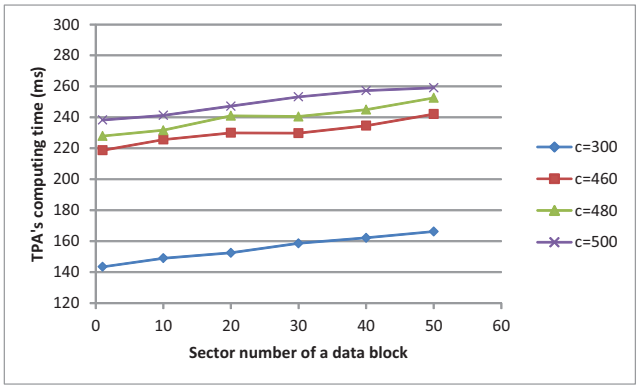


Figure 4: Comparison of computing time for TPA under different s and c

of p is 160 bits. Our test data is a randomly generated 100-MB file. All experimental results represent the mean of 30 trials.

Table 4 presents the experiment result of performance comparison between our scheme and that of [30] under different s and c. It shows that our scheme outperforms the other scheme except for the computing time of *CSP* in the case of $s = 1$. However, as the value of s increases, the *CSP* computing time of [30] will increase significantly because it needs to calculate s pairings during the auditing process. The communication overhead that the *CSP* sends to the *TPA* also increases significantly because the length of an element in group G_T is 120 bytes. Figure 3 and Figure 4 show computing time for *CSP* and *TPA* of our scheme under different s and c. With increase in s and c, the image curves change relatively smoothly and the distance between the curves is relatively uniform. This shows that our scheme is stable and there are no special expensive calculations related to s and c.

Table 3: Storage complexity and communication complexity of different privacy-preserving schemes

Scheme	**User storage complexity**	**Communication complexity of Audit protocol**
[30]	$O(n)$	$O(1)$
[37]	$O(n)$	$O(1)$
Our scheme	$O(1)$	$O(\log n)$

6 Conclusions

In this paper, we proposed a privacy-preserving public auditing scheme with supporting dynamics. The scheme uses the rank-based authenticated skip list as the authenticated search data structure. The formal proof demonstrates that our scheme is secure and has privacy-preserving property in the auditing phase, and performance analysis shows that our scheme is highly efficient.

Acknowledgments

This work is supported by the grants from National Natural Science Foundation of China (No. 61872038). The authors gratefully acknowledge the anonymous reviewers for their valuable comments.

References

[1] Cloud Security Alliance, *Top Threats to Cloud Computing*, 2010. (`https://cloudsecurityalliance.org/topthreats/csathreats.v1.0.pdf`)

[2] G. Ateniese, R. Burns, R. Curtmola, J. Herring, O. Khan, L. Kissner, Z. Peterson, and D. Song, "Remote data checking using provable data possession," *Acm Transactions on Information & System Security*, vol. 14, no. 1, p. 12, 2011.

Table 4: Comparison of computing time for CSP and TPA under different s and c

s=1	**Our scheme**			**[30]**		
Number of sampled blocks c	300	460	500	300	460	500
CSP computing time (ms)	142.28	217.10	233.69	142.37	203.78	227.81
TPA computing time (ms)	143.37	218.66	238.30	147.02	227.75	257.98
s=10	**Our scheme**			**[30]**		
Number of sampled blocks c	300	460	500	300	460	500
CSP computing time (ms)	153.50	227.66	244.85	168.68	245.38	258.29
TPA computing time (ms)	148.97	225.62	241.31	151.89	226.96	255.81

[3] G. Ateniese, R. Burns, R. Curtmola, J. Herring, L. Kissner, Z. Peterson, and D. Song, "Provable data possession at untrusted stores," in *ACM Conference on Computer and Communications Security*, pp. 598–609, 2007.

[4] J. L. Baer and B. Schwab, "A comparison of tree-balancing algorithms," *Communications of the Acm*, vol. 20, no. 5, pp. 322–330, 1977.

[5] A. F. Barsoum and M. A. Hasan, "On verifying dynamic multiple data copies over cloud servers," *Iacr Cryptology Eprint Archive*, vol. 2011, 2011.

[6] D. Boneh and M. K. Franklin, "Identity based encryption from the weil pairing," *Crypto*, vol. 32, no. 3, pp. 213–229, 2001.

[7] R. Curtmola, O. Khan, R. Burns, and G. Ateniese, "Mr-pdp: Multiple-replica provable data possession," in *The International Conference on Distributed Computing Systems*, pp. 411–420, 2008.

[8] B. Dan, B. Lynn, and H. Shacham, "Short signatures from the weil pairing," *Journal of Cryptology*, vol. 17, no. 4, pp. 297–319, 2004.

[9] C. C. Erway, C. Papamanthou, and R. Tamassia, "Dynamic provable data possession," *ACM Transactions on Information and System Security (TISSEC'15)*, vol. 17, no. 4, p. 15, 2015.

[10] M. Etemad and A. Küpcü, "Transparent, distributed, and replicated dynamic provable data possession," in *International Conference on Applied Cryptography and Network Security*, pp. 1–18, 2013.

[11] M. T. Goodrich, R. Tamassia, and A. Schwerin, "Implementation of an authenticated dictionary with skip lists and commutative hashing," in *DARPA Information Survivability Conference & Exposition II*, pp. 68–82 vol.2, 2001.

[12] W. Hsien, C. Yang and M. S. Hwang, "A survey of public auditing for secure data storage in cloud computing," *International Journal of Network Security*, vol. 18, no. 1, pp. 133-142, 2016.

[13] M. S. Hwang, C. C. Lee, T. H. Sun, "Data error locations reported by public auditing in cloud storage service," *Automated Software Engineering*, vol. 21, no. 3, pp. 373–390, Sep. 2014.

[14] M. S. Hwang, T. H. Sun, C. C. Lee, "Achieving dynamic data guarantee and data confidentiality of public auditing in cloud storage service," *Journal of Circuits, Systems, and Computers*, vol. 26, no. 5, 2017.

[15] A. Juels and B. S. Kaliski, "Pors: Proofs of retrievability for large files," in *ACM Conference on Computer and Communications Security*, pp. 584–597, 2007.

[16] J. Katz and Y. Lindel, *Introduction to Modern Cryptography: Principles and Protocols (1 edition)*, British: Chapman and Hall/CRC, 2007.

[17] D. E. Knuth, *The Art of Computer Programming, Volume3: Sorting and Searching (2nd Edition)*, The US: Addison-Wesley Professional, 1998.

[18] Y. Li, Y. Yu, B. Yang, G. Min, and H. Wu, "Privacy preserving cloud data auditing with efficient key update," *Future Generation Computer Systems*, vol. 78, 2016.

[19] C. W. Liu, W. F. Hsien, C. C. Yang, and M. S. Hwang, "A survey of public auditing for shared data storage with user revocation in cloud computing," *International Journal of Network Security*, vol. 18, no. 4, pp. 650–666, 2016.

[20] C. Liu, J. Chen, L. T. Yang, X. Zhang, C. Yang, R. Ranjan, and R. Kotagiri, "Authorized public auditing of dynamic big data storage on cloud with efficient verifiable fine-grained updates," *IEEE Transactions on Parallel & Distributed Systems*, vol. 25, no. 9, pp. 2234–2244, 2014.

[21] W. Mao, *Modern Cryptography: Theory and Practice*, The US: Prentice Hall PTR, 2003.

[22] T. Mell and P. Grance, "Draft nist working definition of cloud computing," *Referenced on*, vol. 53, no. 6, pp. 50–50, 2009.

[23] R. C. Merkle, "A digital signature based on a conventional encryption function," in *Th Conference on Advances in Cryptology*, pp. 369–378, 1987.

[24] C. Papamanthou and R. Tamassia, "Time and space efficient algorithms for two-party authenticated data structures," *Information and Communications Security*, pp. 1-15, 2007.

[25] D. Pointcheval and J. Stern, "Provably secure blind signature schemes," *Proceedings of Asiacrypt*, vol. 1163, pp. 252–265, 1996.

[26] W. Pugh, "Skip lists : A probabilistic alternative to balanced tress," *Communications of the Acm*, vol. 33, no. 6, pp. 668–676, 1990.

[27] H. Shacham and B. Waters, "Compact proofs of retrievability," in *International Conference on the Theory and Application of Cryptology and Information Security*, pp. 90–107, 2008.

[28] M. A. Shah, M. Baker, J. C. Mogul, and R. Swaminathan, "Auditing to keep online storage services honest," in *Hotos'07: Workshop on Hot Topics in Operating Systems*, 2007. (`https://www.usenix.org/legacy/event/hotos07/tech/full_papers/shah/shah_html/hotos11web.html`)

[29] W. Shen, J. Yu, H. Xia, H. Zhang, X. Lu, and R. Hao, "Light-weight and privacy-preserving secure cloud auditing scheme for group users via the third party medium," *Journal of Network & Computer Applications*, vol. 82, pp. 56–64, 2017.

[30] C. Wang, S. S. M. Chow, Q. Wang, K. Ren, and W. Lou, "Privacy-preserving public auditing for secure cloud storage," *IEEE Transactions on Computers*, vol. 62, no. 2, pp. 362–375, 2013.

[31] Q. Wang, C. Wang, K. Ren, W. Lou, and J. Li, "Enabling public auditability and data dynamics for storage security in cloud computing," *IEEE Transactions on Parallel & Distributed Systems*, vol. 22, no. 5, pp. 847–859, 2011.

[32] S. G. Worku, C. Xu, J. Zhao, and X. He, "Secure and efficient privacy-preserving public auditing scheme for cloud storage," *Computers & Electrical Engineering*, vol. 40, no. 5, pp. 1703–1713, 2014.

[33] C. Xu, Y. Zhang, Y. Yu, X. Zhang, and J. Wen, "An efficient provable secure public auditing scheme for cloud storage," *Ksii Transactions on Internet & Information Systems*, vol. 8, no. 11, pp. 4226–4241, 2014.

[34] G. Yang, J. Yu, W. Shen, Q. Su, Z. Fu, and R. Hao, "Enabling public auditing for shared data in cloud storage supporting identity privacy and traceability," *Journal of Systems & Software*, vol. 113, no. C, pp. 130–139, 2016.

[35] J. Yao, S. Chen, S. Nepal, D. Levy, and J. Zic, "Truststore: Making amazon s3 trustworthy with services composition.," in *Ieee/acm International Conference on Cluster, Cloud and Grid Computing*, pp. 600–605, 2010.

[36] Y. Zhu, G. J. Ahn, H. Hu, S. S. Yau, H. G. An, and S. Chen, "Dynamic audit services for outsourced storages in clouds," *IEEE Transactions on Services Computing*, vol. 6, no. 2, pp. 227–238, 2013.

[37] Y. Zhu, H. Hu, G. J. Ahn, and S. S. Yau, "Efficient audit service outsourcing for data integrity in clouds," *Journal of Systems & Software*, vol. 85, no. 5, pp. 1083–1095, 2012.

[38] D. Zissis and D. Lekkas, "Addressing cloud computing security issues," *Future Generation Computer Systems*, vol. 28, no. 3, pp. 583–592, 2012.

Biography

Haichun Zhao received his Bachelor's and Master's degrees in Computer Science and Technology from School of Computer Science at Inner Mongolia Normal University, P. R. China. He is currently a Ph.D. candidate in School of Computer and Communication Engineering at University of Science and Technology Beijing. His current research interests are in information security and cloud computing.

Xuanxia Yao received her B.S. degree from Jiangsu University, M.S. and Ph.D. degree from University of Science and Technology Beijing (USTB), China. She is an associate professor in School of Computer and Communication Engineering, USTB. She is the author of one book, more than 30 articles. Her research interests include network and information security, trusted computing, the security issues in internet of things, cloud computing, blockchain and big data as well.

Xuefeng Zheng is Professor and Ph.D. Supervisor in School of Computer and Communication Engineering at University of Science and Technology Beijing, P.R. China. His research interests include the security of sensor net, cryptosystem and network security.

Authentication Techniques in the Internet of Things Environment: A Survey

Sunghyuck Hong
Division of Communication and Information, Baekseok University
Chungcheongnam-do, Cheonan-si, Dongnam-gu, Anseo-dong 115, Korea
(Email: sunghyuck.hong@gmail.com)
(Received Nov. 2, 2017; Revised and Accepted July 7, 2018; First Online Jan. 30, 2019)

Abstract

The Internet of Things (IoT) enables the offering of services specialized for each client in real time by processing and analyzing the sensor information concerned after storing the sensor information collected from numerous things on the Internet through wired/wireless communications technology. The IoT technology currently applies to various industrial fields including home, medical service, transportation, environmentdisaster, manufacturing, construction, and energy and has been actively researched. The IoT technology closely combined with real life can cause monetary and physical damages to clients by malicious clients through the seizure and falsification of information. Unlike computes and mobile devices, the IoT technology uses micro devices, and therefore lightweight cryptographic techniques with limited storing space are needed. In this regard, this report aims to examine security threatening factors that may occur in the IoT environment and service and to guide authentication techniques by which safe IoT service can be realized. Through this, this research presents a reference guide that can be used by companies or security protocol developers.

Keywords: IoT; Light-Weight Protocol; Secure Authentication; Secure IoT

1 Introduction

1.1 Overview of IoT

IETF, ITUT (MOC), 3GPP (MTC), and ETSI (M2M) defined the security IoT, respectively. Many definitions on the IoT exist, and the IoT concept defined in the CERT-IoT 2009 seems to be most comprehensive and clear. According to it, the IoT is the integrated part of the futuristic Internet and can be defined as a dynamic global network infrastructure equipped with a self-setup function as a mutually compatible communications protocol with standards. Also, CERT-IoT 2009 defines that the IoT consists of self-identifiers, physical things having different characteristics, and virtual things. Upon looking at the definition, things represent a concept containing both physical and virtual things. Here, the examples of virtual things can be software service, software object, and actor as a main player of an act.

Figure 1 shows the concept of IoT [5]. Things, a main component of the IoT, include not only end-devices in the wired/wireless networks, but humans, vehicles, bridges, various electronic equipment, cultural assets, and physical things constituting the natural environment. By expanding the concept of machine to machine (M2M), through which intelligent communications can be conducted between person and machine and between machine and machine using mobile communications network, the concept has evolved into one that can interact with all information in the real and virtual worlds.

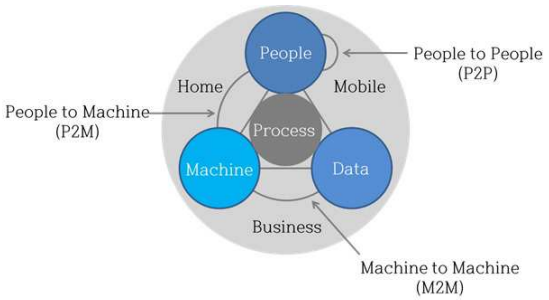


Figure 1: The concept of IoT

1.2 Security-Threatening Factors of IoT

Security threats by component of the IoT are as follows:

- Device layer: This refers to physical threats that include infrastructure paralysis and a threat to life by device suspension and malfunction, information falsification, leakage by the loss, theft, forgery, or falsification of devices, malicious code transition attack thereat between devices, and difficulties to apply IP security technology to lightweight/low power devices.
- Network layer: This refers to information falsification and leakage in the linkage communications process between heterogeneous things networks, damage diffusion due to network and gateway hacking and cross

network devices, and denial of service (DoS) of the IoT by large-scale thingbots.

- Platform service layer: This covers illegal access and attack to platforms by malicious devices/clients, platform collapse according to encryption key hacking after illegal capture, cloud, big data personal information leakage, and privacy infiltration. IoT constitutes specific services by combining various technological factors, such as device, network, and service/platform. Although security technologies to safely protect each technological factor exist, there can be security vulnerability without a method to integrate/link those technological factors.

1.3 Security Technology of IoT

Device security technology needs low power and low weight technologies. Most IoT devices are operated in a low power mode basically and they are low weight devices with low arithmetic operation or storage capacity. Therefore, low power and low weight encryption algorithms considering the performance and required security intensity of low power and low weight IoT devices are needed. Also, a function enduring no-suspension and malfunction, along with a physical threat prevention function, is necessary. Currently a variety of IoT security technologies are being developed worldwide. Low weight cryptographic techniques include PRESENT and KATAN overseas, and LEA and HIGHT domestically. However, the low weight cryptographic techniques developed so far are known to have vulnerability to hacking including auxiliary channel attack. In Korea, ETRI is developing a hardware security module for smartphones [2].

1.4 Network Security Technology

Currently included in the IoT are IEEE 802.15.4 low power communications technology ZigBee as a local area communications technology, IEEE 802.11 wireless LAN technology Wi-Fi whose usage has become universal as it has been used for smartphones, and radio frequency identification (RFID) which is an electronic tag technology adopted for the automatic recognition of things by replacing existing bar codes. As long distance communications technology, 3G or LTE, which are widely used wireless mobile communications modes, are included [3].

- ZigBee: ZigBee has two modes: standard security mode (SSM) for low level security and high security mode (HSM) for high level security. Because each ZigBee device is operated in the open trust mode, reliability of the device is assured. Thus, reliability can be secured if confidentiality and integrity are guaranteed in the communications process between ZigBee devices. However, preparation for a separate security measure is demanded since no encryption is made for all communications sections.Discussion, Implication, and Conclusion;

- Wi-Fi: Wi-Fi is a wireless LAN technology based on IEEE802.11 standard and where high performance wireless communications can be performed. Since communications are conducted in a wireless mode in Wi-Fi, it is especially vulnerable to security threats. If communications data encryption is not conducted in using Wi-Fi, attacks such as wiretapping, snipping, and non-authorized access can occur. As a method to safely protect data in the wireless section, there is a wired equivalent privacy (WEP) authentication protocol presented in the IEEE802.11 standard, and also Wi-Fi protected access (WAP) and WPA 2, which improved WEP's drawbacks, which were proposed in the IEEE802.11i. For security in the access process in addition to the communications process of wireless section, the use of encryption algorithms, such as temporal key integrity protocol (TKIP) and counter mode with CBC-MAC protocol (CCMP), is recommended.

- RFID: RFID is a technology using wireless frequency for the automatic recognition of things and it is based on ISO 18000-7 standard. RFID is a wireless network technology recognizing the tag information attached to things, regardless of physical and visual contacts. It can be the technology receiving most attention recently for the construction of ubiquitous sensor network (USN) environment. Since the passive tag mainly used in the RFID system has limited arithmetic operation capacity and also limitation in the power amount to be used, there is a demerit that high level security technology is difficult to be applied. Being based on wireless communications, RFID is also vulnerable to security threats such as information leakage. As the security requirements in USN, confidentiality and integrity on data communications are required, as well as a safe key control and distribution function. In addition, a safe platform design taking sensor network characteristics into account is needed. Recently, various techniques [8] have been studied for mutual authentication between nodes for data security transmitted in the RFID/USN environment.

- 3G/LTE: Although 3G mobile communications network was a closed structure mainly providing circuit switch, technology-based voice service in the early stage of its introduction, it has gradually evolved into a structure expanding data service in addition to voice, based on packet switching technology. Due to a recent increase in mobile malicious codes, malicious and abnormal traffic of infected terminals is flowing into 3G/4G network. Therefore, a variety of security threats occur. There are many cases in which application of the security equipment used in the existing Internet environment is difficult. Such security equipment has a limitation in that it is difficult to detect or cope with attacks aiming at the unique physical weaknesses of 3G/4G network.

As examined above, security technologies to block attacks, such as infiltration through network or DDoS, are required for IoT service. Studies on the IoT gateway, in order to block infiltration into the IoT network connected with different types of devices and thingbot attack prevention, are evaluated to be at their infant stage. Although Intel, Freescale, and Eurotech developed the IoT gateway, a security control function for the IoT devices and network including real time monitoring or infiltration blocking is not applied.

1.5 Platform/Service Security Technology

The IoT platform/service needs to have an opening structure that can add and develop new services in the mash up type of existing application layer, as well as external sensors or terminals. The relevant studies are actively carried out, centering on the IoT Architecture (IoT-A) project. Mutual authentication, communications encryption, and privacy protection functions between devices, clients, and services need to be provided [16].

As shown in Figure 2, automatic authentication and communications encryption are needed in the IoT without a person's intervention; specifically, a technology offering a new device's service participation registration and key distribution functions to be suitable for service types is necessary. When big data are analyzed by combining various types of sensing information collected in the IoT, a possibility of personal privacy being infiltrated occurs. Therefore, a technology to block automatic personal information collection and non-identification technology are needed in order to prevent personal identification/tracking risk by post-analysis of big data [13].

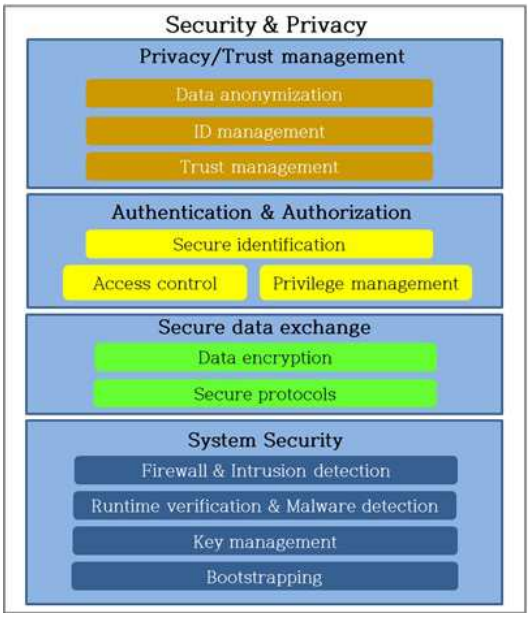


Figure 2: Security technology of IoT service

Although studies on service registration and key control are carried out for IoT equipment authentication, a technology used for user (client) authentication on the existing Internet including PKI is just applied. Even though high growth-applied service fields such as medical service, cars, and home network grow fast, the applied service security technology development is assessed as insufficient.

2 Recommended Encryption Algorithm of Each IoT Component

2.1 Device

A variety of processor platforms are used for the IoT service including 8- and 16-bit processors, which are lightweight devices and 32-bit processor with high performance in terms of IoT devices. Table 1 shows the characteristics of representative processors [6].

To meet confidentiality, integrity, and authentication/authorization, the security requirements in the seven major IoT services (smart home, medical service, traffic, environment/disasters, manufacturing, construction, and energy) should be met by operating symmetric key encryption and open key encryption, which are representative encryption techniques. Concerning symmetric encryption, HIGHT, SEED, ARIA, and LEA (domestic standards), as well as AES (international standard), need to be carried out. Because AES symmetric key encryption offering 128-bit security in the 8-bit processor, for which support is most scarce, requires 0.02ms on average per execution, it is expected to be able to adequately perform encryption/deciphering in the low specs devices.

2.2 Communications and Network

Various processors are used in the IoT devices, while a variety of communications protocols are used as communications stacks. IoT platforms offer various functions, including data control and user (client) and service authentications, so that data sensed from IoT devices can be used by the IoT service. They also have communications interface for communications like Ethernet. As for communications protocols, the IoT platforms support lightweight protocols such as CoAP used in the IoT devices and such protocols as HTTP used in application services. For connection between devices and IoT gateway, wireless communications are mainly used. ZigBee, Bluetooth, Z-Wave, IEEE802.15.4, LoRA, and Wi-Fi are included in the wireless protocols used for the IoT. For the connection between platforms and IoT gateway, REST communications including HTTP and message queuing protocols composed of publication/subscription such as MQTT are used. Between service and platform, protocols such as RPC and SOAP used by mainly API are used so that service can be created through mash-up by the IoT application service [7].

Table 1: Comparison of light and high performance IoT devices

Category	Atmega128	MSP430	ARM-Coriex
Data area	8 bits	16 bits	32 bits
Words	16 bits	16 bits	32 bits
Structure	Harvard	Von Neumann	32 Von Neumann
No. of registers	32	12/16	8/13/16
No. of commands	61	27	56
Core size	6140GE	4913GE	-
Application field	Arduino, Micaz	TelosB	Beagle, Odroid

Table 2 shows standard Encryption Algorithms for Communications/Network.

2.3 Platform

The IoT platform can be an important factor in the IoT environment for data control, client control, and service control and connection between virtual things and physical things. As such, security technology specialized for the platform environment in addition to confidentiality, integrity, and availability, which are the top three security factors in terms of cryptology, should be offered. AllJoyn is an IoT platform developed mainly by AllSeen alliance, centered on Qualcomm. AllJoyn framework consists of Apps and routers. Apps communicate with routers, and the communications between Apps can be conducted through only routers. AllJoyn platform currently supports Bluetooth, Wi-Fi, Ethernet, Serial, and PLC communications, and Zigbee and Z-Wave can be used through bridge software. To offer confidentiality, integrity, and availability (CIA) and authentication in the AllJoyn platform environment, TLS' Pin-code Logon and RSA-based certificate described in RFC5256 are used, and the three factors conduct initial stage communications using SASL. OneM2M recommends 128-bit AES-CBC and AES-CCM to guarantee confidentiality, and recommends the use of HMAC-SHA_256 to assure integrity [9].

3 IoT Authentication Protocol

3.1 ID/Password-based Authentication

This mode serves to store each user's (client's) ID and password in the server's DB, and authentication is carried out on the basis of the stored knowledge. This technology is mainly used in the server/client authentication environment. To prevent authentication disabling due to exposure to the password list stored on the server, there are many cases to adopt a mode storing values through a hash function. To assure higher stability, SSID is hidden, WEP key is used between AP and device, PAP authentication mode is adopted, or RFID mode is used. In the IoT environment, the ID/Password mode has some problems such as server control and load, in light of the IoT environment characteristics where many devices are used without human intervention. Additionally, there is a problem that human intervention should be preconditioned in the device correcting and adding process. The ID/Password mode is not suitable as an authentication technique in the IoT environment since it cannot offer a rejection prevention function [8].

3.2 MAC Address-based Authentication

This is a mode using media access control (MAC) address, which is the identification address allotted to network interface and is mainly used for network access control in the intranet environment. When a device requests access to network, a procedure to authenticate by comparing the MAC address registered in the server and the MAC transmitted with the message requested from the device is undergone. This mode is simpler and more convenient and faster than the ID/Password-based authentication mode. However, there is a need to define a new MAC address style due to the increase of various devices and the advent of IoT, and thus new standards such as EUI-48 and EUI-64 are defined. MAC address is vulnerable to attacks including spoofing due to the absence of separate security equipment as MAC address can be forged [10].

3.3 Code-based Authentication

As a protocol authenticating a thing based on open key codes and symmetric codes, this mode is mainly used for wireless Internet security protocol. This mode supports a variety of standards such as 802.1x/802.11i and WAP. The code protocol-based authentication mode can include such techniques as ID/Password-based authentication, MAC address-based authentication, and certificate-based authentication. By offering various authentication modes, clients can select a suitable authentication mode according to the use environment, and the rejection prevention function is also available according to the adopted code protocol. However, the authentication technique can be connected to vulnerability, if the weakness of code technique is found, because stability is dependent upon code technique [12].

Table 2: Standard encryption algorithms for communications/network

Communications/Network	Support Codes
ZigBee (IEEE 802.15.4)	AES-128
Bluetooth (IEEE 802.15.1)	SAFER-SK128, AES-128
6LoWPAN (IEEE 802.15.4, RFC 4919) hline Z-Wave (IEEE 802.11)	AES-128 RSA-1024, ECC-160 TDES, AES-128
LoRA	AES-128
CoAP (RFC 7252)	AES-128 AES-128 ECC-160
MQTT (OASIS MQTT Version 3.1.1)	AES-128, AES-256, TDES SHA-1, AHA-256, SHA-384
DDS (DDS Security V1.0)	AES-128, AES-256 SHA-1, SHA-256 RSA-2048

3.4 Certificate-based Authentication

Certificate-based Authentication is a mode to authenticate through electronic signature using an open key code. Authentication is carried out on the basis of containing the information for e-signature on the certificate. In Korea, through the E-Signature Act enacted in 1999, the certificate issue system and control regulations were devised. Under the top level certification agency, Root CA, the issue and authentication of certificates are conducted through five authentication (certification) agencies. In foreign countries, personal devices and cable model device authentication, WiMAX industrial certificates through Versign's device authentication service, are offered. In addition, certificate-based authentication technique is used in VoIP and network monitoring cameras, and the areas are gradually expanded.

The certificate-based authentication technique offers high stability through powerful authentication technique, and the reject prevention function is also provided. However, device certificate processing software and algorithms require a high level of arithmetic operation processing volume. Therefore, the technique is not suitable to be used in the low power and low performance IoT devices [13].

3.5 Authentication Using IBE

ID-based authentication is an open key code system using a user's (client's) ID, including email address, name, IP address as an open key, and signature, and where authentication is provided. Although there are such merits as pre-key distribution independence, small arithmetic operation volume, and relatively shorter key length, there is a demerit that ID-based authentication is vulnerable to ID sham attack. As a relatively new concept and technique, compared with other authentication techniques, there are various authentication schemes including Hess's Algorithm, Lunn's Algorithm, and Gentry and Silverberg's Algorithm [14].

3.6 Service-based IoT Authentication Techniques

The IoT environment-based smart home service system consists of home server, home gateway, and smart home devices. As for smart home devices that can be connected with the external Internet through mobile communications network like smartphones, they are limited in a type connected to the external Internet through home gateway [15].

3.7 Requirements of Smart Home Service Security

Table 3 classifies and defines the requirements of smart home service security according to the perspectives of confidentiality, integrity, and availability 3.

3.8 Smart Home Security Function

Smart home means a house or home to which the IoT system monitoring a variety of things and environments, controlling them remotely, or automatically controlling them is applied. The smart home service rapidly grows in the fields including smart home appliances control, cooling/heating control, energy use, HVAC control, crime prevention, and child/baby care. At the CES held in Las Vegas in 2015, smart home based on the IoT gained attraction. Table 4 defines the security function requirements to be applied to home server, home gateway, and home devices consisting of the smart home system. Here, smart home device user (client) authentication is addressed. Smart home devices offer useful information and services to clients through a connection between the service provider and devices within a home. In this process,

Table 3: Classifies and defines the requirements of smart home service

Category	Requirements
Confidentiality	Safe control is needed so that key information for encryption and deciphering used for a user's (client's) important data and encryption algorithms, transmitted and received between smart home devices, cannot be exposed externally. When the data created from a smart home device need to be transmitted externally, they should be transmitted by converting them into a cryptogram type, not in the form of plain text data. The identifier information, by which smart home devices can be identified, should be safely managed so that the information cannot be leaked externally, copied, or modified. Safe and powerful passwords should be set up for smart home devices, and a function to change the set up password cyclically should be provided. In setting up powerful and complex passwords to a home gateway that connects a smart home device with an external communications network, a security function needs to be consolidated.
Integrity	To maintain the reliability and safety of smart home devices, unauthorized devices or unauthorized user (client) access should not be allowed. Safe control is needed so that key information for encryption and deciphering used for a user's (client's) important data and encryption algorithms transmitted and received between smart home devices cannot be exposed externally. Data integrity should be provided when the data created from a smart home device needs to be transmitted externally. Reliable communications environment needs to be shaped through mutual authentication between the devices consisting of smart home service.
Availability	An external attack detection function that can cope with security threats such as cyber attack and hacking should be created. A security function needs to be offered so that smart home device software update can be safely carried out. A device security policy setup function considering various device characteristics and specifications consisting of smart home service should be provided. Smart home devices need to offer a device control system to cope with the theft, loss, addition, and disposition of smart home devices. Smart home device's status needs to be continuously managed and unnecessary remote access should be blocked. Upon detection of abnormal activity of a smart home device, a suitable action should be taken, and the details need to be recorded.

smart home devices can be protected by the security functions of home server and home gateway primarily; however, to cope with security attacks to smart home devices, security functions including compliance with the security grade of each device, hacking prevention, prevention of ID information duplication, and forgery/falsification prevention should be included [16]. A mutual authentication function between smart home devices also needs to be considered so that a data collection and exchange function can be safely conducted between devices in the IoT environment [17].

- Knowledge-based authentication: Knowledge-based authentication is based on what users know, and the passwords and personal identification numbers that users memorize are included. Knowledge-based authentication is based on the secret information shared by a user and the authentication system in advance, and separate devices are not required mostly. Therefore, the cost to build the system is very small, and it is convenient for clients to use, and thus the knowledge-based authentication utilized as a general authentication means a lot. However, there is a drawback in that authentication intensity is weak.

- Possession-based authentication: Possession-based authentication is a mode based on what a user possesses. The characteristic of the possession-based authentication is that users need to use the physical device that the user must possess in the medium used in authentication. Out-of-brand (OOB), OTP token, and open key token authentication are recently used a lot as the authentication means used for possession-based authentication.

- Bio-based authentication: Bio-based authentication is based on the user him/herself and what the user uses, namely, personal physical characteristics classified biologically according to users. Bio-based authentication includes fingerprint, glottis, retina pattern, iris pattern, face shape, palm type, and hand shape. The examples used by clients are signature, which is called behavioral biometrics, and key input pattern perception. Bio-based authentication technique's security is high, and its convenience is also high because it always possesses bio means, compared with possession-based authentication technique. However, system management and construction costs become high because of the high cost of perception devices to perceive body information. Due to such a reason, bio-based authentication is not widely used generally.

3.9 Authentication Techniques for Medical Service and Health Care

The convergence of medical service and ICT is the service including e-health and u-healthcare and has evolved to the mode by which medical service can be offered anywhere and anytime through ICT technology in terms of hospital and treatment environment. Smart healthcare rising recently means a more complex and intelligent level by adding welfare and safety to medical service. As an environment evolves, in which individuals can manage or control exercise amount, meal calories, and sports activity records, the service, provider, and user (client) scopes are expanded [12]. The smart healthcare industry includes hardware such as wearable healthcare devices, and software including healthcare apps, communications and data platforms for healthcare information delivery, and linked medical service.

3.10 IoT – Threats to the Security of Smart Healthcare

- Device hacking: Device hacking refers to the hacking control software that controls hardware, rather than hardware hacking. The control software controlling hardware is generally called firmware. The scope of firmware may mean the OS and also the boot loader uploading the OS to the software. Generally, device hacking means giving the control right of applications to a hacker by hacking OS from various firmware.

- Threats to network security: In the smart healthcare environment where electronic medical devices and wired/wireless network are combined, existing TCP/IP-based security threats and sensor security threats exist.

- Bio-based authentication: Bio-based authentication is based on the user him/herself and what the user uses, namely, personal physical characteristics classified biologically according to users. Bio-based authentication includes fingerprint, glottis, retina pattern, iris pattern, face shape, palm type, and hand shape. The examples used by clients are signature, which is called behavioral biometrics, and key input pattern perception. Bio-based authentication technique's security is high, and its convenience is also high because it always possesses bio means, compared with possession-based authentication technique. However, system management and construction costs become high because of the high cost of perception devices to perceive body information. Due to such a reason, bio-based authentication is not widely used generally.

- Threats to personal information infiltration from service perspective: Security measures for authentication have not been devised on sharing scope, reading/examination restriction, security auditing, and bio information exposure when a patient's personal information and medical information are shared for patient relocation or cooperative treatment. In Korea, incidents that not only leak medical information by hacking, but those that may harm life by capturing hospital information system, occurred (August 2013). A variety of medical information within domestic hospitals was collected by overseas servers, and hackers could manipulate prescriptions randomly by seizing PCs within the hospitals, in addition to medical information leakage.

Because the IoT is used as wearable devices a lot, it needs to be smaller in consideration of design, and there is a limitation that hardware performance becomes lower. Therefore, negligence can occur to security, while basic functions offering can be concentrated. In reality, damages occur gradually due to the IoT device hacking.

3.11 IoT – Information Protection Measures for Smart Healthcare

When communications are conducted with other devices in the IoT environment, the identification and authentication on whether the data are transmitted from the proper devices should be conducted. As for device authentication mode, ID/PW, certificate, and SIM are used. As for ID/PW mode, it is the most basic authentication mode, and separate application and protocol are required for ID/PW authentication between the administrator and device, and personal information needs to be shared in advance. Concerning certificate-based mode, a PKI-based device certificate is widely used. It is safe to use RSA (2,048 bits) and hash function SHA-2 (256 bits) or higher. SIM mode is an authentication mode using USIM or UICC mounted on the terminals, and studies on the mode are actively conducted as communications between devices have been made possible through mobile communications network recently. Since 2011, standardization has been carried out from ETSI and 3GPP starting the embedded SIM Project from GSMA. In Korea, various pilot projects are performed through the participation of mobile carriers such as KT and SKT.

In the IoT, measures to apply TLS, DTLS, IPsec, HIP, and PANA that were applied to the Internet environment to prevent security threats are considered. Specifically, the application of DTLS to a core protocol, CoAP, is set as a basic direction [6]. To apply existing IP-based security protocol, lightweight algorithms are needed in consideration of a device's calculation capacity and memory space. As the method to make DTLS protocol lightweight, there are methods like reducing the number of hand shake messages or simplifying authentication process on the certificate. The initial shaking process is delegated to the owners of resource-limited devices to reduce the hand shaking message packets. The device owners and each device share secret keys safely in advance and form a DTLS session with the server. The device owner encrypts the DTLS ses-

sion information with pre-shared secret key to the device, transmits it, and closes the session. After that, the complex initial stage hand shaking process can be conducted on behalf of the device owner by resuming each session between the device and server. The proposed system, however, has a burden to upload DTLS protocol to the device for DTLS session resumption between the device and server, and thus compressed IPsec is used through various studies [1, 4, 11].

4 Conclusions

The service markets of the IoT are expanded through convergence and compound with various industries. Many studies on security technology in the seven major IoT industrial fields are carried out: Smart home, medical service, transportation, environment/disasters, manufacturing, construction, and energy. The authentication techniques addressed in this report have applied the cryptographic techniques supported in the fields concerned, and they are presumed to be used as baseline data in developing authentication techniques in each field.

Acknowledgments

This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIP) (NRF-2017R1A2B1003394).

References

[1] D. Chatzopoulos and P. Hui, "Asynchronous reputation systems in device-to-device ecosystems," in *2016 IEEE 17th International Symposium on A World of Wireless, Mobile and Multimedia Networks (WoWMoM'16)*, pp. 1–3, June 2016.

[2] F. Corso, Y. Camargo, and L. Ramirez, "Wireless sensor system according to the concept of iot -internet of things-," in *2014 International Conference on Computational Science and Computational Intelligence*, vol. 1, pp. 52–58, Mar. 2014.

[3] D. M. Dobrea and M. C. Dobrea, "Concepts and developments of an wearable system - an iot approach," in *2017 International Symposium on Signals, Circuits and Systems (ISSCS'17)*, pp. 1–4, July 2017.

[4] J. Furtak, Z. Zieliński, and J. Chudzikiewicz, "Security techniques for the wsn link layer within military iot," in *2016 IEEE 3rd World Forum on Internet of Things (WF-IoT'16)*, pp. 233–238, Dec. 2016.

[5] R. Galambos and L. Sujbert, "Active noise control in the concept of iot," in *Proceedings of the 2015 16th International Carpathian Control Conference (ICCC'15)*, pp. 133–137, May 2015.

[6] P. P. Lokulwar and H. R. Deshmukh, "Threat analysis and attacks modelling in routing towards iot," in *2017 International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC'17)*, pp. 721–726, Feb. 2017.

[7] J. Marconot, F. Pebay-Peyroula, and D. Hély, "Iot components lifecycle based security analysis," in *2017 Euromicro Conference on Digital System Design (DSD'17)*, pp. 295–298, Aug. 2017.

[8] S. A. Nauroze, J. G. Hester, B. K. Tehrani, W. Su, J. Bito, R. Bahr, J. Kimionis, and M. M. Tentzeris, "Additively manufactured rf components and modules: Toward empowering the birth of cost-efficient dense and ubiquitous iot implementations," *Proceedings of the IEEE*, vol. 105, pp. 702–722, Apr. 2017.

[9] J. Ni, K. Zhang, X. Lin, and X. S. Shen, "Securing fog computing for internet of things applications: Challenges and solutions," *IEEE Communications Surveys Tutorials*, vol. 20, pp. 601–628, 2018.

[10] S. M. A. Oteafy and H. S. Hassanein, "Resilient iot architectures over dynamic sensor networks with adaptive components," *IEEE Internet of Things Journal*, vol. 4, pp. 474–483, Apr. 2017.

[11] A. Pfitzmann, B. Pfitzmann, M. Schunter, and M. Waidner, "Trusting mobile user devices and security modules," *Computer*, vol. 30, pp. 61–68, Feb. 1997.

[12] K. Rostamzadeh, H. Nicanfar, N. Torabi, S. Gopalakrishnan, and V. C. M. Leung, "A context-aware trust-based information dissemination framework for vehicular networks," *IEEE Internet of Things Journal*, vol. 2, pp. 121–132, Apr. 2015.

[13] H. Shuang and Y. Z. Author, "A study of autonomous method of iot component," in *The 5th International Conference on New Trends in Information Science and Service Science*, vol. 2, pp. 294–298, Oct. 2011.

[14] S. N. Swamy, D. Jadhav, and N. Kulkarni, "Security threats in the application layer in iot applications," in *2017 International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC'17)*, pp. 477–480, Feb. 2017.

[15] A. Syed and R. M. Lourde, "Hardware security threats to dsp applications in an iot network," in *2016 IEEE International Symposium on Nanoelectronic and Information Systems (iNIS'16)*, pp. 62–66, Dec. 2016.

[16] P. Urien, "Introducing tls/dtls secure access modules for iot frameworks: Concepts and experiments," in *2017 IEEE Symposium on Computers and Communications (ISCC'17)*, pp. 220–227, July 2017.

[17] T. Yu, X. Wang, and A. Shami, "Recursive principal component analysis-based data outlier detection and sensor data aggregation in iot systems," *IEEE Internet of Things Journal*, vol. 4, pp. 2207–2216, Dec. 2017.

Biography

Sunghyuck Hong Currently, he is an associate professor in Division of Information and Communication at Baekseok University, and he is a member of editorial board in the Journal of Korean Society for Internet Information (KSII) Transactions on Internet and Information Systems. His current research interests include Blockchain, Secure Mobile Networks, Secure Wireless Sensor Networks, Key Management (Group Key Agreement Protocol), Networks Security (Authentication), Information Security, Embedded Networked Systems, Embedded Software, Wireless LAN, Distributed Systems, Computer Networks, Hybrid Wireless Network Architecture Design, and Mobility Design/Modeling/Simulation. To God, Gloria In Excelsis Deo.

Virus Propagation Behavior Simulation Based on Node Movement Model of Wireless Multi-hop Network

Weimin Kang and Simiao Wang
(Corresponding author: Simiao Wang)

Changchun Medical College, Changchun, Jilin 130062, China
No. 6177, Jilin Road, Changchun, Jilin 130062, China
(Email: simiaow_sm@yeah.net)

(Received May 31, 2018; Revised and Accepted Dec. 17, 2018; First Online Jan. 31, 2019)

Abstract

The improvement of communication technology and the popularity of wireless networks bring us a convenient life, but at the same time they provide a new room for the propagation of viruses, which challenges the security of the Internet. In order to deal with the new kind of virus propagation in wireless multi-hop networks, this study used MATLAB simulation software to simulate SI virus propagation model and random path point model and analyze the effects of the time required for virus transmission, the communication radius of nodes and the number of initial infected nodes on the transmission of SI virus in wireless networks. The result showed that the velocity of node movement could affect the velocity of virus transmission which increased first and then decreased with the increase of node moving speed; the virus transmission speed decreased with the increase of the time required for transmission, and increased with the increase of the communication radius of nodes and the number of initial infected nodes; when the propagation speed was the fastest, the movement speed of the corresponding node was inversely proportional to the time required for propagation, directly proportional to the communication radius, and unrelated to the number of initial infected nodes.

Keywords: Node Movement Model; Virus Transmission Model; Wireless Multi-Hop Networks

1 Introduction

After entering the 21st century, information technology has developed fast, so wireless communication technology gradually replaces wired communication technology [5] and popularizes, especially, the emergence of smart phones, iPad and laptops, which further promote the existence of wireless networks. In wireless multi-hop networks [4], people can obtain resources by using mobile terminals as nodes to access the Internet anytime and anywhere, or as base stations to provide resources. The network is open, besides users and operators, users and users can also provide corresponding services, and they can enter different wireless networks at the least cost; the distribution of the network makes it possible for the network to jump to the rest of the intact nodes normally, even if some of the nodes are missing, to obtain the resources on the Internet. However, it is the open and distributed structure of the wireless multi-hop network [20], which poses a severe challenge to network security. Because of the openness, the network virus camouflaged can freely enter the network, and because of the distributed structure, the transmission of the virus is more complex [2, 8, 11, 12, 16, 19].

In order to guarantee the healthy development of the wireless network, it is necessary to strengthen the defense of the network virus and to study the transmission behavior of the network virus [1, 9, 18]. Lu *et al.* [10] used a random approach to study the evolution and impact of the mobile botnet which was a collection of malicious nodes caused by mobile malware that could perform coordinated attacks. It was discovered that the mobility of a node might be a trigger for a botnet propagation storm. Increasing network bandwidth was an aid to mobile services, but at the same time it was possible to increase the risk that the service was being destroyed by the mobile botnet. Han *et al.* [6] discussed the definition and characteristics of the computer virus, analyzed the virus propagation model and the virus control, proved the necessary and sufficient condition of the non-viral equilibrium point and the local disease equilibrium point of the model by the first method and the disc theorem of the Lyapunov, and then calculated the optimal control strategy of the virus propagation model, which verified the effectiveness of the optimal control.

Sanum *et al.* [15] proposed a mobile Ad Hoc wireless network deterministic node mobility model based on chaos, which realized the deterministic process of ran-

domness generation in mobile mode. The simulation results included the moving path of nodes. Comparing the random speed, direction and average speed of the mobile node under certain control parameters, the method could simulate the mobile mode of real users in a controllable way. In this study, MATLAB simulation software was used to simulate SI virus transmission model and random route point model. The impact of the transmission time, node communication radius and initial infective node number of SI virus on the communication in the network were analyzed.

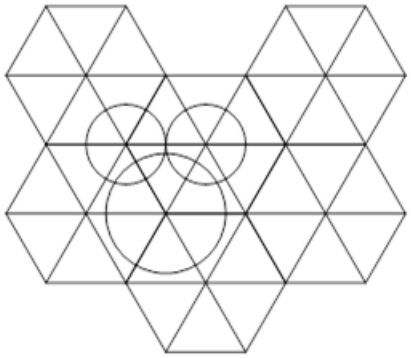


Figure 1: Wireless network topology structure

2 Node Movement Model

Since the beginning of the 21st century, the communication technology has been rapidly improved, from the initial wired communication to the wireless communication which has been widely used, the network constructed by the data interaction is becoming larger and larger, but at the same time the spread of the network virus is also more and more harmful. A wireless multi-hop network consists of multiple nodes, which can be served by a fixed computer or a mobile terminal [3, 14]. Nodes and nodes are connected wirelessly, and data are transmitted to each other through the network. Each wireless node has the function of receiving and transmitting at the same time, and the data in the network jump from one node to the other. Unless all nodes in the network are down, the data must be available to transfer to the target node. The topology of wireless multi-hop networks is not only stable but also extensible. The network node of the original wired communication is limited by the fixed base station, and the network topology of the node [13] will not change, which means the law of the network virus in the transmission process is easy to grasp. However, after the popularization of the wireless communication, the nodes of the wireless network will move with the mobile terminal, and the topology of the network will also change and present irregularity in a short period of time. Therefore, when studying the law of transmission of modern network virus, the movement of nodes needs to be considered.

As shown in Figure 1, the initial network structure in this paper was like that of a cellular network when simulating the propagation law of network viruses. Each intersection point was a network node, and the circle represented the signal coverage range of the node [7]. On the basis of the initial deployment, the node mobility model was added to make the network nodes move within a specified range according to the model rules.

The model of node movement could reflect the law of node movement. The definition of the model could be divided into two types: One was the reproduction of real trajectory, which was reproduced by collecting the moving data of real nodes, but the data that need be processed was too large, which had no extensive practicability; the other was to construct an abstract model, which extracted the motion rules and built a mathematical model by moving data of some real nodes. Random path model (RWP) was one of the most frequently used abstract models in mobile network research. Its principle was Brownian motion. The definition of RWP model [17] was as follows:

1) All nodes were in the rectangular region, and the connection between nodes and nodes was like honeycomb as shown in Figure 1.

2) The nodes moved to any node in the region.

3) The moving speed of the node was within a certain range, and the speed was uniform before reaching the target node.

At the same time, in order to prevent the nodes from leaving the experimental area in the process of random motion, the method of crossing inversion [21] was used to control the motion range and number of nodes. The specific operation was as follows: When any node moves to the boundary of the region, the node was separated from the region from the boundary, and then entered the experimental area again in the symmetrical direction.

3 Virus Transmission Model

The advance of wireless communication technology not only facilitates people's life, but also increases the risk of network security where the network virus is one of the biggest. Its propagation law study can help raise the network security. The transmission mode of network virus in the network is similar to that of the virus in biology. Therefore, according to the characteristics of the transmission of network virus, the transmission model of network virus was constructed with reference to the transmission model of biological virus [25]. At present, there were three common models of virus transmission: SI, SIS and SIR. This paper focused on the SI model [23].

As shown in Figure 2, in the SI model, a node had two shapes: One was the susceptible node "S" (similar to the susceptible population) and the other was the infected node "I" (the equivalent of the source of infection) which represented the probability of a node changing from "S"

to "I" within a certain time range, and its formula [24] was expressed as:

$$\begin{aligned} dS(t) &= -\beta I(t)S(t)dt \\ dI(t) &= \beta I(t)S(t)dt. \end{aligned}$$

$S(t)$ represents the proportion of susceptible nodes at the moment, and $I(t)$ represents the proportion of nodes infected at time. Since the immune mechanism was not set in this model, the infected node would not be re-converted to the easy-to-sense node, and eventually the nodes in the entire network would become Class I nodes.

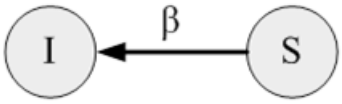


Figure 2: SI virus transmission model

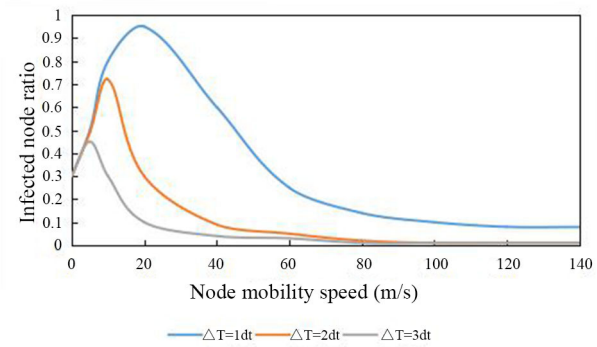


Figure 3: The relationship between node mobility speed and the proportion of infected nodes at time $t = 48s$ under different virus transmission time

4 Simulation Analysis

4.1 Experimental Environment

This paper used the MATLAB simulation platform [22] to compile the network virus propagation model and node moving model. The experiment was carried out on the laboratory server. The server was configured as Windows7 system, I7 processor, 16G memory.

4.2 Experimental Setup

1) Parameter setting of the effect of virus propagation time on the speed of propagation.
Area of node simulation motion region was $\Omega = 1500 \times 1500m^2$; total number of nodes was N=300; number of initial infected nodes was n=3; node communication radius was $r = 30m$; detection time was $t = 48s$. Under the settings of the above parameters, the node infection ratio at different node moving speeds was respectively simulated when the virus transmission time ΔT was 1 dt, 2 dt and 3 dt.

2) Parameter setting of influence of node communication radius on propagation speed.
Area of node simulation motion region was $\Omega = 1500 \times 1500m^2$; total number of nodes was N=300; number of initial infected nodes was n=3; virus transmission time was $\Delta T = 2dt$; detection time was $t = 48s$. Under the setting of the above parameters, the infection ratio of nodes at different moving speeds was simulated when the communication radius of nodes was 20 m, 30 m and 40 m, respectively.

3) Parameter Settings of the influence of the number of initial infection nodes on the propagation speed.
Area of node simulation motion region was $\Omega = 1500 \times 1500m^2$; total number of nodes was N=300; node communication radius was $r = 30m$; virus transmission time was $\Delta T = 2dt$; detection time was $t = 48s$. Under the settings of the above parameters, when the number of initial infected nodes (n) was 3, 6 and 9, respectively, the infection ratio of nodes at different moving speeds was simulated.

4.3 Experiment Results

1) The effect of virus propagation time on the speed of propagation.
As shown in Figure 3, as a whole, as the time was required to spread the virus increases, the speed of infected nodes decreased at the same speed of node movement. The reason was that at a certain node speed, the communication time between nodes could be limited, as the shorter the transmission time was, the greater the probability of successful infection within the communication time and the higher the proportion of infected nodes at would be. Horizontal comparison showed that the speed of the infected nodes increased with the moving speed of the nodes under the transmission time of each virus, which increased first and then decreased. The reason was that with the increase of the moving speed of the nodes, more frequent contact between infected nodes and uninfected ones leaded to an increase in infection rate.

The smaller ΔT was, the faster it would be. When the node moved more than a certain value, although the node contacted more frequently, the communication time between the nodes became shorter, therefore, it was difficult to meet the need of virus transmission time, which made that the probability of infection and the speed of infection decreased. The node moving speed corresponding to the maximum speed of an infected node was called v_{max}. When the time required to spread the virus was 1 dt, v_{max} was

20 m/s; when the time required was 2 dt, v_{max} was 10 m/s; and when the time required was 1 dt, v_{max} was 5 m/s, from which it could be seen that with the time taking to spread the virus, v_{max} decreased because the largest communication times between two nodes were determined by the communication radius and the speed at which the nodes move, so the relationship between the virus spread time and v_{max} was $v_{max} \propto \frac{1}{\Delta T}$.

2) The influence of node communication radius on propagation speed.
As shown in Figure 4, as the node communication radius increased, the infection rate increased at the same node moving speed. The reason was that the increase of communication radius prolonged the communication time between nodes and the virus had more time to complete infection, which leaded to the increase of infection speed.

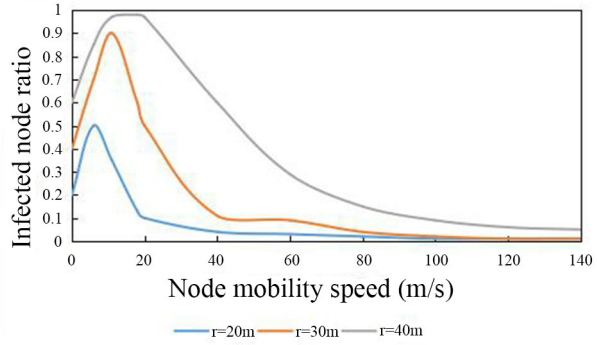


Figure 4: The relationship between node mobility speed and the proportion of infected nodes at time $t = 48s$ under different node communication radius

Horizontally, the speed of infected nodes increased first and then decreased with the increase of node moving speed under every kind of node communication radius. As mentioned above, the increase of node mobility increased the connectivity of nodes. The infection probability increased, the communication time decreased and the infection probability decreased when the speed was too large. At the same time, the larger the communication radius were, the faster the infection speed increased but the slower it decreased. When the communication radius was 20/m, v_{max} was 6 m/s; when the communication radius was 30/m, v_{max} was 11m/s; when the communication radius is 40/m, v_{max} was 18 m/s, and it could be found out that with the increase in the radius of communication, v_{max} increased which was because of an increase in the radius of the virus, and even if the speed of movement increased, it could be guaranteed that the virus would complete the transmission when the communication time was sufficient, so that the relationship between the node communication radius and v_{max} was $v_{max} \propto r$.

3) The effect of initial infective nodal points on the speed of transmission.
As shown in Figure 5, with the increase of initial infected nodes, the infection speed increased with the same node moving speed. The reason was that the increase of initial infected nodes increased the frequency of contact between infected nodes and susceptible nodes, and the probability of successful transmission of the virus increased, which leaded to the increase of infection speed.

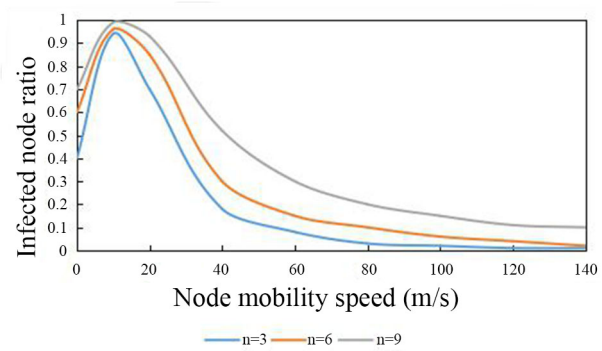


Figure 5: The relationship between node mobility speed and the proportion of infected nodes at time $t = 48s$ under different initial number of infected nodes

Transverse comparison showed that with the increase of node moving speed, the infection node velocity increased first and then decreased with the number of initial infected nodes. As the reason was mentioned above, the more the initial infected nodes were, the faster the infection velocity increased and the slower the infection speed decreased. Whether the initial number of infected nodes was 3, 6 or 9, the corresponding v_{max} were all 10 m/s, from which it could be noticed that v_{max} remained steady with the change of initial infected nodes. The reason was that the infection probability of nodes was related to the network connectivity and the time required to spread the virus. As the network connectivity was an inherent attribute of the network, it was only related to the communication radius and the moving speed of the nodes, and whether the nodes were infected or not would not affect the network connectivity. The transmission time was the attribute of the virus itself, and the nodal infection would not affect the virus itself. In this experiment, the communication radius and propagation time were fixed, so v_{max} would have no change.

5 Conclusion

In this study, SI virus transmission model and RWP model were simulated by MATLAB simulation software, and the effects of virus transmission time, node communication radius and initial infective node number on the

transmission of SI virus in wireless network were analyzed. The results showed that:

1) The infection speed of the virus in the network increased first and then decreased with the increase of the moving speed of the nodes.

2) With the increase of the time required for virus transmission, the infection rate of the virus in the network decreased.

3) With the increase of the node communication radius, the infection speed of the virus increased, and the infection speed was the most rapid.

4) With the increase of the initial number of infected nodes, the infection speed of the virus increased, and the corresponding node movement speed did not change at the fastest speed.

In this paper, the biological virus transmission model is applied to the transmission of network virus, and the RWP model is used to simulate the movement of people. Compared with the analysis method of real trajectory reconstruction, this method requires less computation and has higher accuracy.

References

[1] D. S. AbdElminaam, "Improving the security of cloud computing by building new hybrid cryptography algorithms," *International Journal of Electronics and Information Engineering*, vol. 8, no. 1, pp. 40-48, 2018.

[2] M. H. R. Al-Shaikhly, H. M. El-Bakry, and A. A. Saleh, "Cloud security using Markov chain and genetic algorithm," *International Journal of Electronics and Information Engineering*, vol. 8, no. 2, pp. 96-106, 2018.

[3] A. A. Al-khatib, W. A. Hammood, "Mobile malware and defending systems: Comparison study," *International Journal of Electronics and Information Engineering*, vol. 6, no. 2, pp. 116–123, 2017.

[4] S. Chawla, M. Manju, S. Singh, "Computational intelligence techniques for wireless sensor network: Review," *International Journal of Computer Applications*, vol. 118, no. 14, pp. 23–27, 2015.

[5] H. Chen, X. Wang, "Research on network evolution model based on computer virus transmission mechanism," in *International Conference on Informative & Cybernetics for Computational Social Systems*, IEEE, 2016.

[6] C. Han, L. Li, "Control on the transmission of computer viruses in network," *Automatic Control & Computer Sciences*, vol. 51, no. 4, pp. 233–239, 2017.

[7] J. He, K. Ma, Y. Yang, "Research on wireless network topology of substation equipment monitoring based on QoS," in *International Conference on Intelligent Computation Technology and Automation*, IEEE, pp. 605–608, 2015.

[8] S. Islam, H. Ali, A. Habib, N. Nobi, M. Alam, and D. Hossain, "Threat minimization by design and deployment of secured networking model," *International Journal of Electronics and Information Engineering*, vol. 8, no. 2, pp. 135–144, 2018.

[9] M. Kumar, K. Dutta, I. Chopra, "Impact of wormhole attack on data aggregation in hieracrchical WSN," *International Journal of Electronics and Information Engineering*, vol. 1, no. 2, pp. 70–77, 2014.

[10] Z. Lu, W. Wang, C. Wang, "On the evolution and impact of mobile botnets in wireless networks," *IEEE Transactions on Mobile Computing*, vol. 15, no. 9, pp. 2304–2316, 2016.

[11] 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.

[12] 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.

[13] M. K. Priyan, G. U. Devi, "Energy efficient node selection algorithm based on node performance index and random waypoint mobility model in internet of vehicles," *Cluster Computing*, vol. 4, pp. 1-15, 2017.

[14] A. Rana, D. Sharma, "Mobile ad-hoc clustering using inclusive particle swarm optimization algorithm," *International Journal of Electronics and Information Engineering*, vol. 8, no. 1, pp. 1-8, 2018.

[15] W. Sanum, P. Ketthong, J. Noymanee, "A deterministic node mobility model for mobile ad hoc wireless network using signum-based discrete-time chaotic map," in *Telecommunication Networks & Applications Conference*, IEEE, pp. 114–119, 2015.

[16] R. K. Shakya, "Modified SI epidemic model for combating virus spread in spatially correlated wireless sensor networks," *eprint arXiv:1801.04744*, Jan. 2018.

[17] R. T. Silva, R. R. Colletti, C. Pimentel, *et al.*, "BETA random waypoint mobility model for wireless network simulation," *Ad Hoc Networks*, vol. 48, pp. 93–100, 2016.

[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] A. Tayal, N. Mishra and S. Sharma, "Active monitoring & postmortem forensic analysis of network threats: A survey," *International Journal of Electronics and Information Engineering*, vol. 6, no. 1, pp. 49–59, 2017.

[20] L. Wang, C. Yao, Y. Yang, *et al.*, "Research on a dynamic virus propagation model to improve smart campus security," *IEEE Access*, vol. 6, pp. 20663, 20671, 2018.

[21] X. Wang, W. Ni, K. Zheng, *et al.*, "Virus propagation modeling and convergence analysis in large-scale networks," *IEEE Transactions on Information Forensics & Security*, vol. 11, no. 10, pp. 2241–2254, 2016.

[22] Y. Xu, J. Ren, "Propagation effect of a virus outbreak on a network with limited anti-virus ability," *PlosOne*, vol. 11, no. 10, pp. e0164415, 2016.

[23] L. Yang, M. Draief, X. Yang, "Heterogeneous virus propagation in networks: A theoretical study," *Mathematical Methods in the Applied Sciences*, vol. 40, no. 5, 2017.

[24] C. Zhang, H. Huang, "Optimal control strategy for a novel computer virus propagation model on scale-free networks," *Physica A: Statistical Mechanics & Its Applications*, vol. 451, pp. 251–265, 2016. Volume 451, 1 June 2016, Pages 251-265

[25] J. Zhang, M. Yang, D. Zhang, "The research of virus propagation and immunization strategy based on the restricted edge weighted BBV network model," in *International Symposium on Computational Intelligence and Design*, IEEE, pp. 204–209, 2016.

Biography

Weimin Kang, born in June 1972, graduated from Jilin University with a master's degree. Her research direction is computer application technology, and she is currently working as an associate professor at Changchun Medical College. She published Computer Basic Examination Reform in Medical Colleges in 2013, Computer Application Fundamentals and Test Analysis in 2014, Opinions on the Development of Health Information Management in China, and Design of the Expert System for Computer Rank Examination Based on J2EE in 2015, Application of Information Retrieval and Information Statistics in Hospital Information System in 2016, How to Use the Use of Disease Codes in Medical Records to Unify Medical Charges in 2017 and Study on the Practical Teaching System of Health Information Management Based on Post Competence in 2019. She held the subject of Computer Level Examination Tutoring Teaching Expert System Based on Artificial Intelligence Technology in 2014, Application Research of "2+3" High-level Talents Training Program for Health Information Management Major in 2016 and international disease classification coding and surgical coding in the application of medical records in 2017.

Simiao Wang, born in 1989, graduated from Indiana University with a master's degree. Her research direction is health information management, and she now works at Changchun Medical College as an assistant. She participated in the subject of Application Research of the "2+3" High-level Talents Training Program for Health Information Management Major in 2016 and Application Research of International Classification of Diseases and Surgical Coding in Medical Record Management in 2017. She published Review of the Application and Review of DRGs in China and the United States and the Status and Necessity of Integrating EBM Research Evidence into EHR in China and the United States in 2017.

A Cloud Computing Oriented Neural Network for Resource Demands and Management Scheduling

Gaoxiang Lou and Zongyan Cai
(Corresponding author: Gaoxiang Lou)

School of Construction Machinery, Chang'an University
Xi'an, Shaanxi 710064, China
(Email: agoxzl@126.com)

(Received May 31, 2018; Revised and Accepted Dec. 17, 2018; First Online Jan. 31, 2019)

Abstract

Cloud computing, a new kind of resource sharing service system, can provide virtual resource services such as infrastructure and platform for users who access it through the Internet. Its service quality is related to resource management and scheduling. In this study, CloudSim3.0 simulation platform was used as a simulation platform for cloud computing resource scheduling to test the performance of radial base function (RBF) neural network based on particle swarm optimization (PSO) and RBF neural network based on Improved Particle Swarm Optimization (IPSO) in cloud resource scheduling and configuration. The results showed that the CPU and memory utilization rate and processing time of the two algorithms increased with the increase of processing tasks. It was found that compared to PSO-RBF, IPSO-RBF had higher CPU and memory utilization rate and shorter processing time and converged faster and found the best position of particles after only 30 iterations with small fluctuation amplitude. In addition, IPSO-RBF had better performance in balancing the load of different kinds of physical resources compared to PSO-RBF.

Keywords: Cloud Computing; Particle Swarm Algorithm; Radial Base Function Neural Network; Resource Scheduling

1 Introduction

With the development of society, the role of computer in various fields of society is becoming more and more extensive. At the same time, the demand for computing power in industries that need to use computer power is also increasing day by day [6]. Single improvement of computer hardware performance to obtain more computing power not only has limited computing power improvement, but also has low cost performance for a single user compared with the cost of hardware improvement [9].

The emergence of cloud computing solves the problem of limited performance improvement of a single computer. Relying on the Internet, cloud computing uploaded and distributed the tasks that users needed to process to the "master station" composed of a large number of servers, and applied the physical resources in the "master station" to process the tasks of users [1, 3]. Its "master station" is called the resource pool [15], also known as "cloud". Cloud computing combines virtualization, parallel computing, distributed computing and other concepts, and has the following characteristics [5].

Cloud computing has a larger computing scale than a single user; Cloud computing can complete the interaction of information through the Internet and terminals, and the resources it uses are not physical objects [10, 13]. Resource pools in cloud computing are Shared. Relevant researches are as follows. Chen et al. [4] proposed an Improved Ant Colony System (IACS) method and conducted extensive experiments based on workflows of different scales and different cloud resources. Experimental results showed that IACS was able to find a better solution with lower cost than basic particle swarm optimization (PSO) and dynamic target genetic algorithm under different scheduling scales and deadlines. Abdullahi Mohammed et al. [11] proposed a symbiotic organisms search optimization algorithm (SOS) based on simulated annealing (SA) to improve the convergence speed and quality of SOS solutions.

CloudSim simulation results showed that SASOS was superior to SOS in terms of convergence speed, response time, unbalance and MAK. Zhang et al. [18] proposed an improved Centrino Hardware Control (CHC) algorithm, which inherited the advantages of standard genetic algorithm (SGA) and CHC algorithm. The experimental results showed that the improved CHC algorithm had better efficiency and convergence, and the average time and completion time of task scheduling were relatively shorter. In this paper, CloudSim3.0 simulation platform was adopted

as the simulation platform for cloud computing resource scheduling to test the scheduling configuration performance of RBF neural network based on PSO and RBF neural network based on improved particle swarm optimization (IPSO).

2 Cloud Computing Resource Scheduling

The resource scheduling model in cloud computing consists of three layers: service request, virtual resource pool and physical resource pool. Cloud computing resource scheduling was generally divided into two steps. First, the task in the service request was allocated to the virtual machine in the virtual resource pool, and then the allocated virtual machine was deployed to the physical machine in the physical resource pool, as showed in Figure 1.

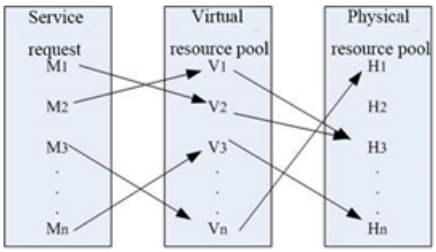


Figure 1: Cloud computing resource scheduling

Its mathematical model [2] was: If the task in the service request was divided into n mutually independent subtasks, the set of sub-tasks was $M = \{m_1, m_2, \cdots, m_n\}$, where the i-th sub-task was m_i, and a sub-task could only run in one virtual node. If there were b virtual nodes $(b < n)$, the set of virtual nodes was $V = \{v_1, v_2, \cdots, v_b\}$, where v_j is the j-th virtual node. Then the distribution could be expressed by the matrix A:

$$A = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \cdots & \cdots & \cdots & \cdots \\ a_{b1} & a_{b2} & \cdots & a_{bn} \end{bmatrix} \quad (1)$$

Any element in the matrix represents the relationship between the sub-task and the virtual section. When the i-th task run on the JTH node, it was 1, otherwise it was 0. For the convenience of later calculation, it was assumed that a node could only run one task at a time, and a task could only run on one node at the same time. Hence the time spent in completing task M was:

$$time = \max(\sum_{i=1}^{n} a_{ji} t_{ji}, \quad (1 \le i \le n, 1 \le j \le b). \quad (2)$$

A good resource scheduling algorithm should have the least time to complete the task, *i.e.*,

$$Cost = \min(time). \quad (3)$$

Equations (1), (2) and (3) were the mathematical model of cloud computing resource scheduling.

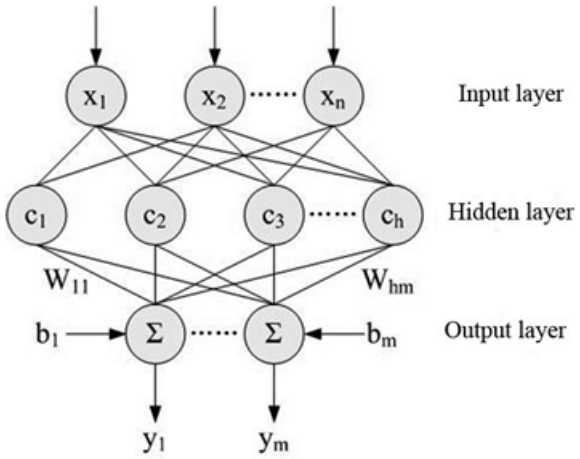


Figure 2: Basic structure of RBF network

3 Radial Basis Function (RBF) Neural Network

Radial basis function neural network [**?**, 19] has a simple structure. It converges quickly and could simulate any nonlinear function. Its structure is shown in Figure 2. RBF neural network belongs to feed-forward neural network, and its basic structure was divided into three layers, including input layer, hidden layer and output layer. The activation function of hidden layer is radial basis function, so each node of hidden layer has a data center c_i.

Each data node in the input layer was denoted as $x = (x_1, x_2, \cdots, x_i, \cdots, x_n)^T$. The output data in the output layer was denoted as $y = (y_1, y_2, \cdots, y_i, \cdots, y_n)^T$. The mapping between the hidden layer and the output layer was a linear mapping, and the weight matrix of the mapping between the nodes of the two layers was denoted as $w = (w_{11}, w_{12}, \cdots, w_{ij}, \cdots, w_{hm})^T$. The mapping from an input layer to a hidden layer in RBF network is a nonlinear mapping, and its mapping formula [7] was:

$$h_i = \exp(\frac{||x - c_i||^2}{2b_i^2}), \quad 1 \le i \le h, \quad (4)$$

where h_i refers to the output of the i-th node in the hidden layer, $||\cdot||$ is the euclidean distance, and b_i refers to the width of the radial basis function of the node of the i-th hidden layer. The mapping formula between the hidden layer and the output layer was:

$$y_j = \sum_{i=1}^{h} w_{ij} h_i, \quad 1 \le j \le m, \quad (5)$$

where y_j is the output data of the j-th node in the output layer and w_{ij} is the mapping weight between the i-th hidden layer node and the j-th output layer node.

4 RBF Network Based on PSO

PSO [17] is also known as "flock foraging algorithm" because it is derived from the research on the foraging and migration behavior of birds. Its principle is to obtain the global optimal solution by tracing the current optimal solution. PSO is one of the evolutionary algorithms. Similar to other evolutionary algorithms, PSO uses population to find the solution set in space, and iterates randomly on population initialization and end conditions to get the optimal solution. In the process of operation, the direction of the optimal solution was determined by the fitness value, and the fitness value was used to judge whether the solution was good or not. The global optimal solution was found by comparing the self-optimal solution with the currently explored optimal solution.In particle swarm optimization, individual updates were influenced by historical particles rather than random ones. The number of particles in the particle swarm optimization algorithm was n. In m-dimensional space, P_i was used as the vector position of the i-th particle, i.e., $\overrightarrow{P_i} = (p_{i1}, p_{i2}, \cdots, p_{iM})$, S_i as the velocity of the particle $\overrightarrow{V_i} = (v_{i1}, v_{i2}, \cdots, v_{iM})$.

Let the current optimal position of the i-th particle be $pbest_i$, the optimal position of the particle swarm was $gbest_i$. Then the formula [16] for the velocity change of the particle was:

$$V_{i+1} = \mu V_i + a_1 x_1 (pbest_i - P_i) + a_2 x_2 (gbest_i - P_i), \quad (6)$$

where μ refers to the value of the particle affected by inertia, a_1 and a_2 are the learning factor, x_1 and x_2 are a random number, they were evenly distributed between 0 and 1, $a_1 x_1 (pbest_i - P_i)$ is cognitive term, and $a_2 x_2 (gbest_i - P_i)$ refers to social term.

The expression of the position change of the particle was:

$$P_{i+1} = P_i + V_i. \quad (7)$$

By repeatedly updating the calculation based on Equations (6) and (7), and analyzing the fitness of particles, the optimal solution with the largest fitness could be found. This study adopted PSO algorithm to optimize RBF network. First, the mapping parameters in the RBF network were converted into the dimensional vectors of each particle in the PSO. In other words, mapping parameters such as data center c_i, width coefficient b_i, and weight w_{ij} in RBF network were taken as dimensions in corresponding particles. Then, the fitness function in PSO was taken as the mean square error in RBF network, and the optimal weight with the minimum mean square error could be obtained after PSO operation.

After the mapping parameters in RBF were converted into the dimensions of particles in the particle swarm, the optimal parameters of a set of particles were calculated according to the calculation flow in Figure 3, and then the dimensions of the particles were converted into the mapping parameters in the RBF network to participate in the scheduling and configuration of cloud resources by the RBF neural network.

5 RBF Network Based on Improved Particle Swarm Optimization

Up to now, there are various ways to improve the traditional particle swarm optimization algorithm, but the ultimate purpose is to make up for the two shortcomings of the traditional particle swarm optimization: First, when the test object is a complex function, with the increase of the number of iterations, the algorithm is likely to fall into a local extreme value, which is difficult to obtain the real optimal solution. The second is the selection of algorithm parameters, among which the inertia factor and learning factor have the greatest influence on the change of algorithm capability.

In the traditional PSO algorithm, cognitive coefficient a1 and social coefficient a2 remain unchanged, they are learning factors; as a result, when the number of iterations is small, the influence of individual cognition is large; when the number of iterations is large, the social influence is large, failing to reflect the aforementioned changes at the same time, which makes the algorithm obtain the local optimal solution and induces the phenomenon of "premature" [14]. In order to solve the above problems, the traditional particle swarm optimization algorithm was improved, and the improved formula [12] was:

$$S_{i+1} = \mu V_i + b(\frac{1}{n}) x_1 (pbest_i - P_i) + cn^2 x_2 (gbest_i - P_i). \quad (8)$$

By comparing Equations (6) and (8), it was found that the improvement made is to change the cognitive term coefficient a_1 into $b(\frac{1}{n})$, so that the individual cognitive proportion decreased with the increase of the number of iterations. The social item coefficient a_2 changes to cn^2, so that the social item proportion of the group increases with the number of iterations. Then, when calculating, b took a large value, and c took a small value, making it conform to the fact that individual cognition was the dominant factor in the initial stage. As the number of iterations increases, the proportion of social terms increases, and it started to become the dominant factor.

After the improvement, the algorithm was more consistent with the objective law of the optimal solution, and the practicability was greatly improved. The optimization steps of the IPSO for RBF network were not much different from those of PSO for RBF above. Similarly, the mapping parameters in RBF were converted into the dimensions of particles in the particle swarm. Then, the optimal parameters of the particle swarm were calculated according to the steps shown in Figure 3, and the difference was that the formula for updating the particle velocity is replaced by Equations (6) with (8). Then, the dimensions of the optimal particle swarm were converted into various mapping parameters in the RBF network, and participate in the scheduling and configuration of cloud resources by the RBF neural network.

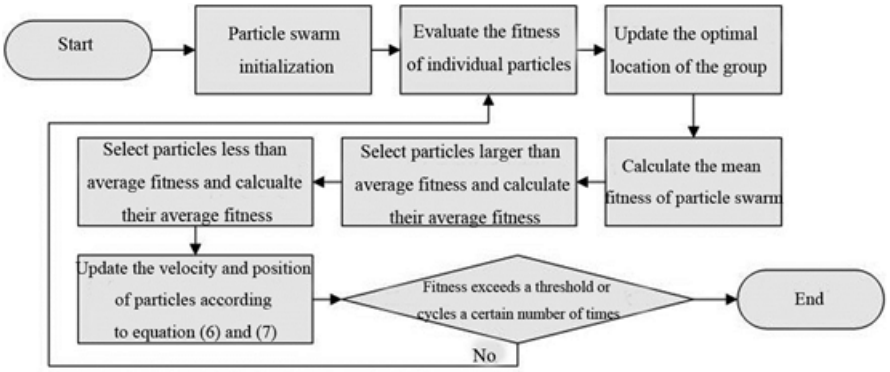


Figure 3: Calculation flow of RBF network mapping parameters based on PSO algorithm

6 Experimental Analysis

6.1 Experimental Environment

In this study, CloudSim3.0 simulation platform [8] was selected as the simulation platform for cloud computing resource scheduling to test the scheduling configuration performance of PSO-based RBF neural network and IPSO-based RBF neural network for cloud resources.The experimental environment in this paper was Windows10 operating system on hardware with 16G of memory and 1000G of hard disk storage. On the software, CloudSim3.0 cloud platform simulation simulator, compilation environment JDK1.7 and development tool MyEclipse were adopted.

6.2 Experiment Settings

Thirty physical resources were set up in the cloud computing laboratory center and converted into virtual machine resources. The configuration of each virtual machine was 2GB memory capacity and 3.0 GHz CPU processing frequency. Virtual physical resources were divided into three parts: responsible for processing document classes, for image processing, and for dealing with video class, and the maximum number of iterations was set as 100. The learning factor was set as 1.33. The number of submitted tasks was set as 100, including three types: 10 types of documents, 30 types of pictures, and 60 types of video. Each algorithm was iterated for 100 times, and the experiment was repeated for 40 times. The average value of each test result was obtained.

6.3 Experimental Results

As showed in Figure 4, under the premise that the total number of cloud resources was constant, the utilization rate of CPU and memory of the two algorithms increased accordingly with the increase of the number of tasks executed; when the number of tasks exceeded a certain number, the utilization rate was relatively stable; the CPU utilization of PSO-RBF increased with the number of tasks before executing 50 tasks, and fluctuated around 0.4 after more than 50 tasks; the CPU utilization rate of IPSO-RBF increased with the number of tasks before 60 tasks were executed, and fluctuated around 0.6 after more than 60 tasks. For memory utilization, PSO-RBF fluctuated around 0.4 after 80 tasks and IPSO-RBF fluctuated around 0.5 after 60 tasks. On the whole, both CPU utilization and memory utilization rates were higher compared with IPSO-RBF, because PSO-RBF algorithm used more physical resources, tasks were evenly distributed to each virtual machine node, and the utilization of CPU and memory was evenly distributed.

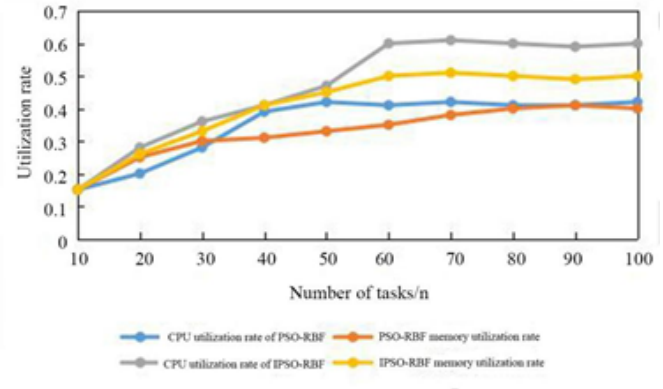


Figure 4: CPU and memory utilization rates of the two algorithms under different task number

As showed in Figure 5, with the increased of the number of processing tasks, the time required by the two algorithms for processing tasks also showed an upward trend of fluctuation. The fluctuation range of PSO-RBF was larger, and the time required to process the same number of tasks was greater than those of IPSO-RBF in most cases. It could be seen from the figure that it took the most time to process 70 tasks in this experiment, which took 910 ms. The time required for IPSO-RBF task processing fluctuated with the increase of the task, and the fluctuation range was small. It was found that it took the most time, 80 ms, to process 100 tasks in this experiment.

On the whole, IPSO-RBF was more efficient and took less time to process tasks.

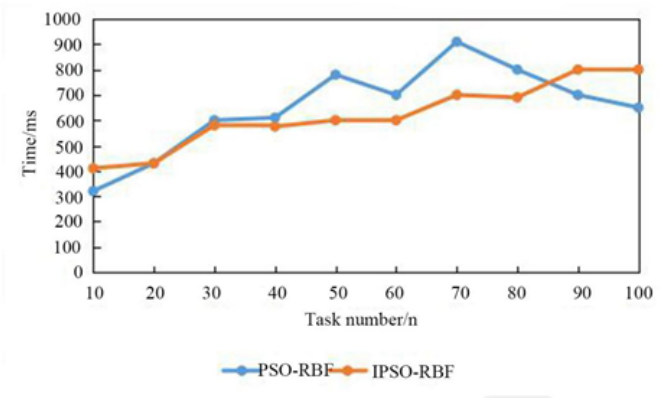


Figure 5: Time consumed by the two algorithms in processing different number of tasks

As showed in Figure 6, with the increase of iteration times, the convergence of the two algorithms became stable. In the process of convergence, the particle position of PSO-RBF increased with the number of iterations and fluctuated greatly. It did not stabilize until 90 iterations. The particle position of IPSO-RBF increased with the number of iterations, with a small fluctuation range, and the position of the optimal solution was found stably after 30 iterations. IPSO-RBF had a fast convergence speed and a better effect of finding the optimal solution.

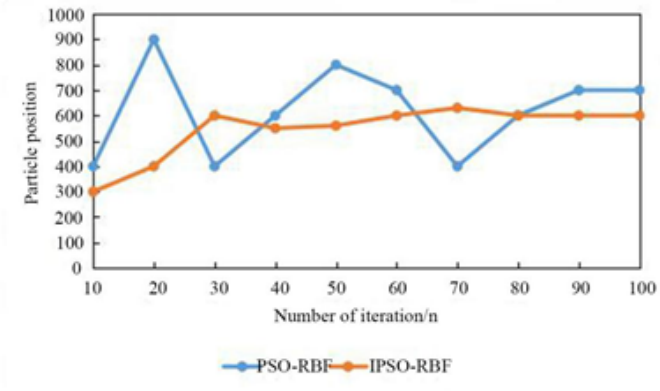


Figure 6: Convergence effect of the two algorithms

As showed in Figure 7, due to the difference in processing capacity between nodes, the number of tasks assigned on different types of physical resource nodes was different. In the PSO-RBF algorithm, the number of tasks processed on the three types of nodes was similar, and the number of actual task types was compared. It could be found that the PSO-RBF algorithm basically distributes tasks to all nodes on an equal basis without considering the difference in processing capability of different types of tasks between different nodes. However, the number of tasks on different types of nodes allocated by IPSO-RBF algorithm was obviously different. By comparing the number of actual types of tasks, it was found that this algorithm considers whether the type of tasks was consistent with the type of resource nodes in the allocation of tasks, and balanced the load between different nodes.

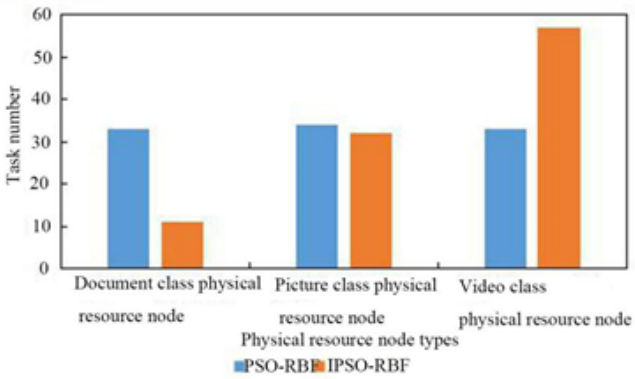


Figure 7: Load distribution on nodes of different physical resource types

7 Conclusion

Firstly, this paper briefly introduced the cloud computing resource scheduling model and the construction of RBF neural network. In cloud computing resource scheduling, tasks were allocated to virtual machine according to requests, and then virtual machine was allocated with physical resources. Then, the PSO-based RBF neural network algorithm and IPSO-based RBF neural network algorithm were proposed. Finally, simulation experiments were carried out on the CloudSim3.0 simulation platform for the two algorithms. The results showed that the CPU and memory utilization rate and running time of the virtual machine increased with the increase of the number of tasks. Moreover, IPSO-RBF algorithm had a higher utilization rate of virtual machine resources and shorter running time than PSO-RBF algorithm. In terms of the convergence effect, IPSO-RBF algorithm could achieve stable convergence faster with only 30 iterations, while PSO-RBF algorithm needed 90 iterations to achieve stable convergence. IPSO-RBF could balance the load of different kinds of physical resources better than PSO-RBF.

References

[1] D. S. AbdElminaam, "Improving the security of cloud computing by building new hybrid cryptography algorithms," *International Journal of Electronics and Information Engineering*, vol. 8, no. 1, pp. 40–48, 2018.

[2] A. Alexandridis, E. Chondrodima, H. Sarimveis, "Cooperative learning for radial basis function networks using particle swarm optimization," *Applied Soft Computing*, vol. 49, pp. 485–497, 2016.
[3] M. H. R. Al-Shaikhly, H. M. El-Bakry, and A. A. Saleh, "Cloud security using Markov chain and genetic algorithm," *International Journal of Electronics and Information Engineering*, vol. 8, no. 2, pp. 96–106, 2018.
[4] Z. G. Chen, Z. H. Zhan, H. H. Li, *et al.*, "Deadline constrained cloud computing resources scheduling through an ant colony system approach," in *International Conference on Cloud Computing Research and Innovation*, IEEE, pp. 112–119, 2015.
[5] M. C. S. Filho, R. L. Oliveira, C. C. Monteiro, *et al.*, "CloudSim Plus: A cloud computing simulation framework pursuing software engineering principles for improved modularity, extensibility and correctness," in *Integrated Network and Service Management*, IEEE, pp. 400–406, 2017.
[6] M. B. Gawali, S. K. Shinde, "Task scheduling and resource allocation in cloud computing using a heuristic approach," *Journal of Cloud Computing*, vol. 7, no. 1, pp. 4, 2018.
[7] S. Guo, J. Liu, Y. Yang, *et al.*, "Energy-efficient dynamic computation offloading and cooperative task scheduling in mobile cloud computing," *IEEE Transactions on Mobile Computing*, vol. 18, no. 2, pp. pp: 319–333, 2018.
[8] H. Han, X. Wu, L. Zhang, *et al.*, "Self-organizing RBF neural network using an adaptive gradient multiobjective particle swarm optimization," *IEEE Transactions on Cybernetics*, vol. 49, no. 1, pp. 69–82, 2017.
[9] N. Kumar, N. Chilamkurti, S. Zeadally, *et al.*, "Achieving quality of service (QoS) using resource allocation and adaptive scheduling in cloud computing with grid support," *Computer Journal*, vol. 57, no. 2, pp. 281–290, 2018.
[10] L. Liu, Z. Cao, C. Mao, "A note on one outsourcing scheme for big data access control in cloud," *International Journal of Electronics and Information Engineering*, vol. 9, no. 1, pp. 29-35, 2018.
[11] A. Mohammed, N. Md Asri, "Hybrid symbiotic organisms search optimization algorithm for scheduling of tasks on cloud computing environment," *PLoS One*, vol. 11, no. 6, pp. e0158229, 2016.
[12] A. D. Niros, G. E. Tsekouras, D. Tsolakis, *et al.*, "Hierarchical fuzzy clustering in conjunction with particle swarm optimization to efficiently design RBF neural networks," *Journal of Intelligent & Robotic Systems*, vol. 78, no. 1, pp. 105–125, 2015.
[13] S. Rezaei, M. Ali Doostari, and M. Bayat, "A lightweight and efficient data sharing scheme for cloud computing," *International Journal of Electronics and Information Engineering*, vol. 9, no. 2, pp. 115-131, 2018.
[14] K. Shanmukhi, P. R. Vundavilli, B. Surekha, "Modeling of ECDM micro-drilling process using GA- and PSO-trained radial basis function neural network," *Soft Computing*, vol. 19, no. 8, pp. 2193–2202, 2015.
[15] W. Wei, X. Fan, H. Song, *et al.*, "Imperfect information dynamic stackelberg game based resource allocation using hidden markov for cloud computing," *IEEE Transactions on Services Computing*, vol. 11, no. 1, pp. 78–89, 2018.
[16] J. Wu, J. Long, M. Liu, "Evolving RBF neural networks for rainfall prediction using hybrid particle swarm optimization and genetic algorithm," *Neurocomputing*, vol. 148, no. 2, pp. 136–142, 2015.
[17] L. Xu, F. Qian, Y. Li, *et al.*, "Resource allocation based on quantum particle swarm optimization and RBF neural network for overlay cognitive OFDM System," *Neurocomputing*, vol. 173, no. 3, pp. 1250–1256, 2016.
[18] L. Zhang, W. Tong, S. Lu, "Task scheduling of cloud computing based on Improved CHC algorithm," in *International Conference on Audio, Language and Image Processing*, IEEE, pp. 574–577, 2015.
[19] W. Zhang, D. Wei, "Prediction for network traffic of radial basis function neural network model based on improved particle swarm optimization algorithm," *Neural Computing & Applications*, vol. 29, no. 1, pp. 1–10, 2016.

Biography

Gaoxiang Lou (1989-). Ph.D. candidate from Chang'an University. His research interests include Mixed flow assembly scheduling, Manufacturing system integration and automation, etc. Funding institutions: National Natural Science Foundation, China(51305042),Construction project of Central University Educational Reform Special Foundation China(jgy16049,jgy170501). Recently published papers: Research on the Hybrid Algorithm based on Differential Evolution and Genetic Algorithm for Mixed Model Assembly Scheduling, Improved hybrid immune clonal selection genetic algorithm and its application in hybrid shop scheduling, etc.

Zongyan Cai (1964-). Doctor of Education , Professor. Worked in Chang'an University. His research interests include Manufacturing system integration and automation, Intelligent robot technology, etc. Funding institutions: National Natural Science Foundation, China(51705030),Construction project of Central University Educational Reform Special Foundation China(jgy16049,jgy170501). Recently published papers: An enhanced bearing fault diagnosis method based on TVF-EMD and a high-order energy operator, An alternative demodulation method using envelope-derivative operator for bearing fault diagnosis of the vibrating screen, etc.

Trust in Ad Hoc Networks: A New Model Based on Clustering Algorithm

Ali Mansouri[1] and Mohamed Salim Bouhlel[2]
(Corresponding author: Ali Mansouri)
Higher Institute of Applied Languages and Computer Sciences of Beja[1]
Boulevard of the Environment B.P. 340 Beja 9000, Tunisia
Higher Institute of Biotechnology of Sfax (ISBS)[2]
Soukra Road km 4, B.P. 1175, 3038 Sfax, Tunisia
(Email: mehermansouri@yahoo.fr)
(Received Nov. 13, 2017; Revised and Accepted May 5, 2018; First Online Feb. 16, 2019)

Abstract

An Mobile Ad-hoc network (MANET) is formed when group of mobile wireless nodes collaborate between them to communicate through wireless links in the absence of the fixed infrastructure and any centralized control. This paper focuses on the design a self-stabilizing clustering algorithm in MANETs. The Topology that we propose is a partitioning based on trust between members of a group. It forms a structure able to adapt dynamically to changes in the topology. Some cryptographic-based schemes have been proposed to secure the clustering process, but they are unable to handle the internal attacks. To defend against insider malicious nodes, trust and reputation management systems should be used .Our solution is based on our efficient trust model and distributed algorithm to clustering network.We present our clustering approach based on trust for applications in the field of security.

Keywords: Clustering Algorithm; Maintenance of the Topology; Mobile Ad Hoc; Self-stabilizing; Trust Relationships

1 Introduction

A mobile ad hoc network is a collection Mobile entities interconnected by a wireless network forming temporary independently of any infrastructure or centralized administration. The nodes in Mobile ad hoc networks join and leave the networks dynamically. At some point of time there is a possibility of enormous increase in the size of the network. Handling nodes in big network may put a burden on network management schemes and may introduce delays in the net-work.

Dividing big networks in small groups called clusters may prove to be a good solution for handling them in a better and efficient manner. As MANET (Mobile Ad hoc networks) is self organized, the challenge of achieving security is critical. Evolving and managing trust relationships among the nodes in the network are important to carry efficient transmissions Clustering organize the ad hoc networks hierarchically and create clusters of ad hoc nodes which are geographically adjacent. Each cluster is managed by a cluster head (CH) and other nodes may act as cluster gateway or cluster member.

In this article, we present a clustering approach for efficient, scalable and secure clustering of MANETs. Our proposal consists on forming clusters around the trustworthy nodes; in other words, the node that has highest trust value is elected as the cluster head. A threshold of trustworthy is used to perform system stability.

2 Related Work

In the literature, there are many proposals to construct clusters in mobile ad hoc networks.The first algorithms of Lowest-ID clustering algorithm (LID) proposed by Baker and Ephremides [5]. Clustering High-Connectivity (HCC) of are based on a particular criterion the selection of cluster-heads, which is the identifier of a node. This algorithm to form clusters in a single jump, where each member is its direct neighbor cluster-head. In the construction phase clusters, nodes communicate with their neighbors to have a local knowledge and thus fix the cluster-head.

This phase is repeated periodically for any topology change. The algorithm cited in [15]is a modified version of the Lowest-ID algorithm .The authors propose a clustering algorithm to reduce the work-Clustering fic control. A node broadcasts a single message containing his clustering decision. According to his local knowledge of the topology, each node decides to become a head-cluster or not. This decision is communicated to the neighborhood, forcing the neighbors of the new cluster-head who are not yet affiliated a cluster choose it as a cluster-head.

In [6], authors propose energy efficient secure trust based clustering algorithm for mobile wireless sensor network. Their solution creates one hop members to minimize the overhead and take into account the trust level of a node, mobility, remaining energy and its distance to neighbors.

In [18], authors present a preference-based protocol for trust and head selection for cluster-based MANET perform the tasks of a certification authority and proactive secret sharing scheme is used to distribute the private network key to the CHs. In this solution, each cluster is first formed based on the trust values of the neighbor nodes. To create cluster, an ad hoc node evaluates its neighbor nodes' of neighbor nodes; each node chooses one node that has the highest value as its trust guarantor. Then, the chosen node becomes the CH and the chooser becomes a member of the cluster, a node of the second highest trust value is chosen, in this way, a cluster is formed by the CH which has the highest trust value among the cluster members.

The other trust-based clustering scheme is designed in [1], Authors propose trust based secure on demand routing protocol (TSDRP) for Manet's. In this scheme each node evaluates the trust value of neighbor nodes and recommends one of neighbors that have the highest trust value as its trust guarantor. Then a node becomes a member of CH node which is one-hop away.

In [16], authors propose a self-stabilizing clustering algorithm in mobile ad hoc networks Clustering Algorithm is another trust-based clustering scheme. It evaluates the stability of node through computing the neighbor change ratio and the residual battery power of mobile nodes. To elect CHs by using the voting mechanism, each node votes other nodes only if the node is the most trustful one among its neighbor nodes and the node's stability is better than itself.

In [10], authors propose an efficient secure group communication in MANET using fuzzy trust based clustering and hierarchical distributed group key management which includes a trust value defining how much any node is trusted by its neighborhood and used the certificate as node's identifier. It uses voting mechanism to elect the most trusted node.

In [17],authors give a honey bee algorithm–based efficient cluster formation and optimization scheme in mobile ad hoc networks. It aims to elect trust worthy stable CHs that can provide secure communication via cooperative nodes. The authors in [17], authors propose performance analysis of TSDRP and AODV routing protocol under black hole attacks in Manets by varying network size.

The authors in [12], authors present a multi-metric-based algorithm for cluster head selection in multi-hop ad hoc network to improve the search performance and scalability of MANETs with trust mechanism. In this solution, the trust relationship is formed by evaluating the level of trust using Bayesian statistic analysis and clusters can be formed and maintained with only partial knowledge which makes it suitable for distributed autonomous MANETs.

In [11], authors give a model of mobility aware clustering scheme with bayesian-evidence trust management for public key infrastructure in ad hoc networks.

The authors in [14], propose an efficient trust-based scheme for secure and quality of service routing in MANETs. A composite trust model for secure routing in mobile ad-hoc networks is proposed in [13], and Trust threshold based public key management in mobile ad hoc networks proposed in [3]. Also, a preference-based protocol for trust and head selection for cluster-based MANET is proposed in [18].

3 Global Architecture and Criteria of Clustering

This section introduces our topology. We assume first that all nodes periodically broadcast a hello message to their neighbors in a single jump for the information of the nodes around them. Our topology is organized clustered. Each cluster consists of a cluster-head, a core and a periphery.

- The cluster-head is the node that identifies the cluster. He is responsible for the communication between clusters. The cluster-head is the root of a under tree built during the clustering process and covers all members the cluster.
- The core is the center of cluster. The cluster-head is one of the members of the core of its cluster. All core members are neighbors to the cluster-head.
- The periphery is composed of cluster members that are not in the core. Figure 1 illustrâtes the main features and elements of our topology.

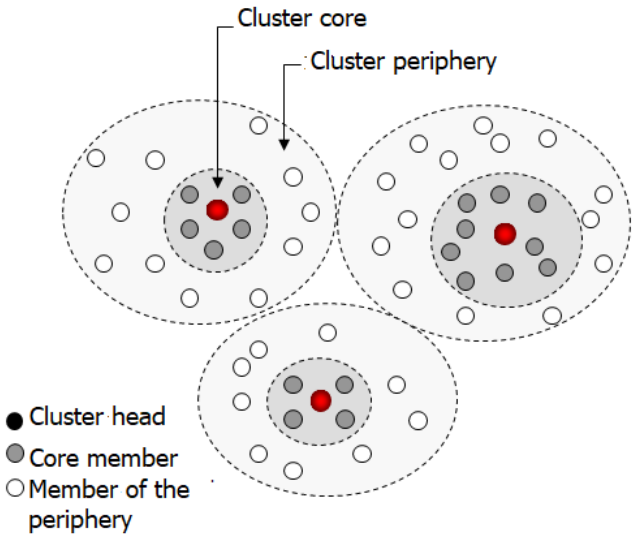


Figure 1: General structure of our topology

We are interested in the clustering area of security. Our algorithm uses trust as partitioning criteria.Trust is fundamental to maintain a certain level of security. This is a important aspect in the design of network.

However, in a dynamic and mobile environment without trusted authority centralized, it is not easy to assess confidence. Many existing solutions propose to calculate the confidence in MANET based on the information that a node can collect other nodes passively [8].

As these interactions are frequent in the cooperative behavior of the nodes of a MANET, it will not difficult to quickly establish a first estimate of the level of trust between direct neighbors. If ordinary interactions are not sufficient to evaluate a ratio trust/distrust, the nodes can generate additional traffic to evaluate how much they trust their neighbors. Thus, to manage its trust, each node i maintains tv value of trust (i, j) for any neighbor j to node i (and possibly former neighbors and nodes not adjacent where the node i receives recommendations). This value reflects the degree of trust or distrust node i has on its neighbor j. Several trust functions were proposed in the literature. We use quantification confidence proposed in [9] and we extend to reflect Account trusted recommendations developed in [20].

Thus, our confidence values are real numbers between -1 and +1. A number negative represents the degree of distrust. -1 indicates a total distrust. A number positive represents the degree of confidence. 1 represents absolute confidence. When a new or unknown node j between in the neighborhood of node i, the node i initializes $tv(i, j)$ to a first value init__trust ($tv(i, j) = \text{init\_\_trust}$).

This initial value is useful for two nodes that have not yet reported together. For example, if they are completely unknown $init\_\_trust = -1$ else $init\_\_trust = 1$.

Note that two neighboring i and j may have different interpretations of their exchanges.Thus, $tv(i, j)$ may be different from $tv(j, i)$. We use the following function to calculate the value of the confidence node i to node j:

$$tv(i,j) = \tanh\left(\sum_{k=1}^{n_1} u_k w_k\right) \quad (1)$$

Where n_1 is the number of interactions between the two nodes. w_k is the weight of the interaction number k. u_k is 1 if the k is positive interaction and -1 if it is negative. The function tanh is used to project the sum of different interactions in the interval [-1, 1].

Several examples of interactions can be used to calculate confidence values . All these interactions is based on the routing information. Here we develop the all interactions we use in our clustering algorithm.

Passive Knowledge: A node can obtain important information a neighbor in road construction, for example. In fact, if a node starts in "promiscuous" after the transmission of all packets to hear the retransmission by the destination node, it can get the following information about this neighbor [2]:

- It acts as a black hole if the packet is not forwarded.
- There is a change of attack if the content has changed.
- It makes an attack if a manufacturing self-produced packet is transmitted.
- It makes an identity theft attack if the IP addresses were falsified.
- It induces delays by delaying the retransmission of the packet.

Accuracy/packet error: When a node receives a correct packet, can increase the value of trust which he attributes to the one that sent the packet, and other nodes in the path from the source (if the protocol routing provides information on the nodes that make up a route from a source to a destination). Similarly, if the received packet is wrong, the receiver can reduce the value of the confidence he attributes to the sender of the package and the value of the confidence he attributes to intermediate nodes of the road.

Altruistic Behavior: If an intermediate node on a route to a destination given, receives a packet for which the next hop is not available, it can remove the package and notify the sender. Even So, if there is a route to the final destination can use this route from its cache, send the packet on the new road and notify the sender the broken link. If the new road is to be correct, it reveals that the sender Error in altruistic behavior. Therefore, this information can be used to increase trust between the two nodes. Compliance / non-compliance with the rules of clustering: a node that does not meet the clustering rules is obviously a malicious node. His neighbors may detect this problem by observing how it sets its clustering variables, described later in this chapter in its hello messages.

Inconsistent Trust: A node that distributes false reports trust or lies about his relationship of trust is malicious. This behavior may be detected by comparing the ratio of trust receipt and monitoring communications of its neighbors.

Communication with malicious nodes: when a node regularly exchanges messages with a malicious node, it is considered a malicious node. These direct assessments of trust can be strengthened by reports distributed trust. The confidence reports allow nodes to share the information in confidence and disseminate in the network. A simple approach to distribute the relationship of trust is for each node to broadcast only trusting relationships with its immediate neighbors. A report confidence initiated by node k lists the values of trust that has the other nodes, namely $tv(k, j)$. When node i receives reports of confidence of a certain node j, it uses them to improve their confidence value $tv(i, j)$ as follows:

Where $n2$ is the number of nodes that have sent confidence reports node j to node i. If the received one report of an un-known node, report trust is not considered. In addition, the use of $tv(i,k)$ as a weight for a relationship of trust initiated by a node k favors direct considerations confidence. Some malicious nodes can lie about trust. However, these false reports can be detected by neighbors.Monitoring Mutual nodes avoids inconsistencies trusted reports received.

To use the confidence values calculated by the nodes as a criterion for clustering,we define two confidence thresholds S_{min} and S_{max} where $S_{min} \leq S_{max}, S_{max} \in [0,1]$ and $S_{min} \in [-1,0]$.

- Full Trust (TT): A relationship between two nodes i and j is a relationship full trust $((Relation(i,j) = TT) if tv(i,j) \in [S_{max},1] and tv(j,i) \in [S_{max},1])$.
- Partial Trust (PT): A relationship between two nodes i and j is a relationship partial trust (Relation (i,j) = PT) if and only if:
  - $Tv(i,j) \in [S_{max},1]$;
  - $Tv(j,i) \in [S_{max},1]$;
  - $Tv(i,j) \in [S_{min},S_{max}]$.
- Suspicion (DT): A relationship between two nodes i and j is a relationship distrust $((Relation(i,j) = DT)$ if and only if $tv(i,j) \in [-1,S_{min}]$ or $tv(j,i) \in [-1,S_{min}])$.

Note that the three relationships are symmetrical. For example, if a node i has a total trust with a node j then the node j has a relationship total confidence with node i. In the following, we develop the different steps of our algorithm clustering.

4 Clustering Algorithm Based on Trust

This algorithm uses a distributed heuristic and tries to minimize the explicit information of the formation of clusters. Our algorithm is composed of three sub algorithms: 1, 2 and 3. In the follows, the authors show the rules forming algorithm.

Algorithm 1 :RULE0: The excluded members.

Begin
if $\vee j \in Ni.Relation(i,j) = DT$ **then**
$CH_i = null \wedge Parent_i = null.$
end if

4.1 Description of Rules of the Algorithm

In this section, we present the rules of our clustering algorithm.We start by characterizing a legitimate state. A legitimate state is stable clustering training cluster-heads, core members, members of the periphery and members excluded. That algorithm consists of five detailed rules , each node determines the role as hello messages it receives from its neighbors.

Algorithm 2 :RULE1 & RULE2: Selection and updating the cluster-head members

RULE1
Begin
if $\vee j \in Ni.Relation(i,j) = DT$ **then**
$CH_i = null \wedge Parent_i = null.$
end if
RULE2
if $\vee j \in N.i[Relation(i,j) = TT \wedge Compare(TT - edge_i, TT - edge_j)] \wedge \vee j \in N.i[CH_j \neq j \wedge (CH_j = null \wedge Relation(i,j) = TT) \wedge Compare(TT - edge_i, TT - edge_j)])$ **then**
$CH_i = i \wedge Parent_i = i \wedge Hop - CH_i = 0$
end if

Algorithm 3 :RULE3 & RULE4: Selection and update the other cluster nodes.

RULE3
if $\vee k \in N.i \exists j \in N.iCompare(Relation(i,j), Relation(i,k)) \wedge CH_i \neq i$ **then**
$CH_i = CH_j \wedge Parent_i = j \wedge HopCH_i = HopCH_{j+1}$
end if
RULE4
if $CH_i \neq CHParent_i \vee HopCH_i \neq HopCHParent_{i+1}$ **then**
$CH_i = null \wedge Parent_i = null \wedge HopCH_i = null$
end if

In our algorithm, the rules (R0), (R2) and (R4) have priority over other rules (R1) and (R3). Indeed, an asset that has a top incorrect value of its variable initializes its state null. Then, it executes the rule corresponding to become a cluster-head, a core member, a member of the periphery or excluded member. The clustering algorithm added to hello messages the following fields:

- $TT - edge_i$: Number of TT relationships a node i have with its neighbors.
- CH_i: The cluster cluster-head is attached node i. CH_i is equal to null if node i does not yet belong to a cluster.
- $Core_i$: The kernel member belongs node i. If i is a clusterhead or ring member then $Core_i$ is set to i. If i is not yet attached to a kernel then $Core_i$ is set to null.
- $Hop - CH_i$: Number of hops from node i to clusterhead.
- $Tv(i,j)$: For each neighbor j, this field represents the value of the trust from node i to node j.

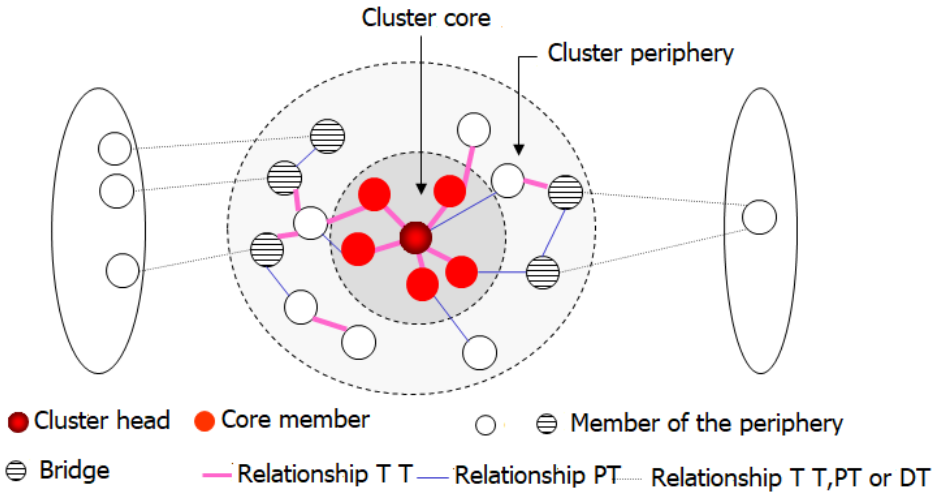


Figure 2: The resulting clustering

- $Parent_i$: This field expresses the father of node i in the cor-responding subtree. Clustering is performed in three fully dynamic and distributed phases:

  Phase 1: Election of cluster-head. Each node which has the largest number TT-edge among its neighbors who do not yet belong to a cluster declares as a cluster-head. In case of a tie the node with the most neighbors is elected. In case of equal values TT-edge and the equal number of neighbors, the node that has the greater identity is privileged. Once a node becomes a cluster-head, he puts his identity in the scope of its CH hello messages and changes the value of Hop field - CH to zero.

  Phase 2: Core training. All the neighbors who have relation-ships TT with a cluster-head form the cluster core. Their hello messages contain the identity of the cluster-head in the CH field and the value 1 in the Hop-CH field. A node can have relationships with several TT cluster-heads. In this case, the node chooses the cluster-head that has the largest TT-edge.

  Phase 3: Formation of the periphery of the core. After the phase 1 and Phase 2, the nodes surrounding the core joining the cluster according to the two steps following (TT privileged relationship is a relationship that PT is privileged to DT relationship.

  Step 1. Members of the periphery having TT relations. After incorporation cores, if any of the nodes that have not yet ad-hered to a cluster TT and have relations with at least one ring, they join the cluster which they have the greatest confidence value.

  Step 2. Members of the periphery having PT relationships. The latter step is to add the nodes that share relationships with the PT least one node in the cluster. A node in this category favors cluster with which it has the lowest distance (number of hops) to the cluster-head. The neighbor with whom he has a relationship PT and the lowest distance clusterhead became his father in the sub-tree rooted. The cluster-head is the root of the subtree. This subtree simplifies communication between clusters. A node in the periphery with at least one neighbor belonging in another cluster is called a gateway. When a node joins a cluster, it updates the CH and CH-Hop fields of hello message.

Clustering obtained is shown in Figure 2.

To succeed clustering, despite the presence of malicious nodes, the honest nodes cooperate closely. They do not communicate the message clustering malicious nodes and ignore all messages from clustering these nodes. Thus, clustering messages and data dissemination spend only by TT relation-ships or relationships PT. However, even if all malicious nodes were detected, clustering can be disrupted.

The condition on the number of malicious nodes and their dispersion in the network is necessary. In fact, if the network is not sufficiently dense and malicious nodes are scattered so that they prevent honest nodes to participate in clustering, the protocol will fail to achieve a complete clustering. As a result, isolated nodes and clusters can appear disconnected.

4.2 Clustering Example

We explain our clustering algorithm by applying it to the set of nodes described in Figure 3. In this example, we assume that the nodes have already calculated their confidence values from their direct neighbors. Each node has a unique identifier and is denominated by the trust he attributes to his neighbors. The confidence thresholds are set at $S_{min} = 0$ and $S_{max} = 0.2$.

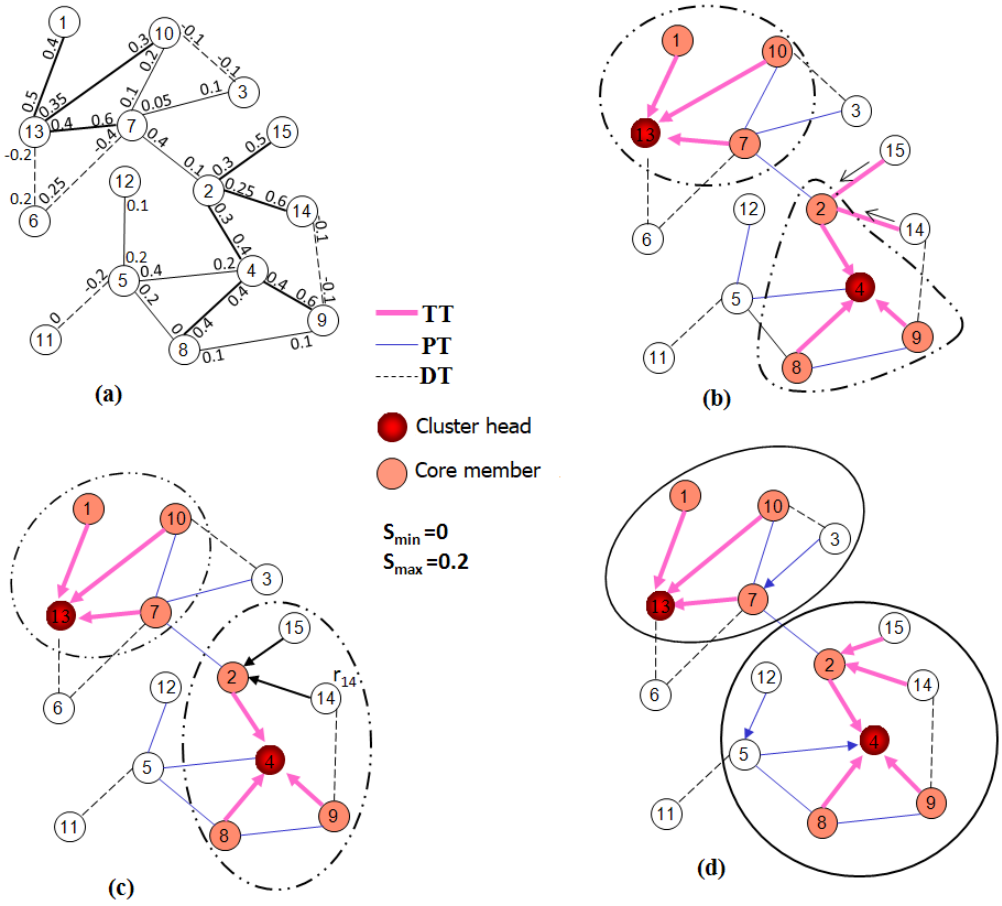


Figure 3: Example of clustering

Figure 3(a) illustrates the different relationships TT, PT and DT in the network according to the values of S_{min} and S_{max}. Figure3(b) show nodes 4 and 13 are clusterheads according to the number of TT relationships they have with their neighbors.

Recall that this information is broadcast in hello messages. Each of the nodes 4 and 13 puts its identity in the CH field and updates the Hop-CH field to 0. Then, nodes 1, 7 and 10 (respectively 2, 8 and 9) which share a relation TT with cluster-head 13 (respectively 4) form the core of it (see Figure 3 (c)) Nodes 1, 7 and 10 (respectively 2, 8 and 9) change their fields CH and Hop - CH from their hello messages to respectively 13 and 1 (respectively 4 and 1). The two nodes 14 and 15 have TT relations with node 2. Thus, they join cluster 4. They are attached to kernel member 2 (see Figure 3(c)).

They update their CH and Hop-CH fields accordingly. In the last step, the nodes 12 and 5 (respectively 3) that share a PT relationship with a node belonging to cluster 4 (respectively 13), join this cluster and update the CH and Hop - CH fields of their hello messages. Oriented edges in the example illustrate the subtrees built during the clustering process. When a node joins a cluster, it chooses a father in the shortest way to the cluster-head. Nodes 6 and 11 do not share any TT or PT relationships with other nodes of the network. So, the clustering algorithm does not take into account these nodes in the clusters obtained.

4.3 Convergence and Accuracy of the Algorithm

We show in this section that our algorithm converges to a state legitimate. For example, from any initial state, the algorithm con-verges to a stable state composed of cluster-heads, core members, member of the periphery and excluded members. We will assume for this that the nodes are not malicious.

Lemma 1. *The node running the rule (R0) does not change its state only time and remains stable there after.*

Proof. A node that only sharing DT relations with its neighbors will belong to any cluster, and runs the rule (R0). In addition, any of its neighbors will consider it, and it will be ignored. Therefore, it will no longer change state. □

Lemma 2. *The node running the rule (R1)stabilizes after more* $(\triangle - 1)^2$ *movements.*

Proof. The node that has the maximum number of TT compared Relations with all its neighbors with which it shares a TT relationship will become a cluster-head. It does not change state thereafter because the decision is

local and has best value (the maximum number of TT relations). This comparison depends only the number of relationships that TT is previously updated by the message hello. Against by, in some cases, the rule (R1), the node is declared as a cluster-head because it has the maximum number of relationships TT compared with all neighbors with whom it shares TT relationship and who are not yet to a cluster. His decision depends only on its neighbors which are stabilized at the after $(\Delta - 1)$ move. Hence, a cluster-head stabilizes more after $(\triangle - 1)^2$ movements. □

Lemma 3. *The system enters a legitimate state in O* $(n(\triangle - 1)^2)$.

Proof. The other vertices which are not cluster-heads always choose the parent with whom they share the strongest relationship. This Parent exists and is unique because the Com-pare function (x, y) selects a single vertex. The best relationship is the relationship with a cluster-head, which is the root of subtree. According to the previous lemma, a cluster-head converges to a stable state more after $(\triangle - 1)^2$ movements. The algorithm converges and a legitimate state $O(n(\triangle - 1)^2)$. □

5 Maintenance of the Topology

The clustering algorithm is self-stabilizing. It runs continuously and readjusts clusters based on trust relationships between nodes. Relationships confidence evolves over time depending on the interactions between the nodes. The Mobility can also change the situation of clusters. In fact, when a node acquires new neighbors or loses some of them, because of mobility, several changes can appear in the situation of the node inside Cluster:

1) The number of TT (TT-edge) relationships of the node can change. For example, if the node is a cluster-head, it can no longer be. If the node is a member from the periphery, it can become a member of the kernel or a cluster-head.

2) The PT or TT relationships that link a node to a cluster can break and the node no longer belongs to the cluster.

3) A node that has only DT relationships with its neighbors can acquire PT or TT relationships and thus joins a cluster.

4) Etc. Phase 1, Phase 2 and Phase 3 of the clustering algorithm are re-executed as often as necessary to form new clusters or up-date existing clusters.

5.1 Election of a New Cluster-head

Several situations may involve the election of a new cluster-head. We focus on two of these situations: the failure of the current cluster-head and the change the TT-edge value of the current cluster-head so that it is no longer the node with the highest TT- edge value among its neighbors.

1) Cluster-head failure: If a cluster-head goes down, its wires (that's to say other members of the same kernel) no longer receive his hello messages periodicals. In this case, the kernel members re-initialize the CH field from their messages hello to null. Upon receipt of these modifications of the message hello, members of the underlying periphery are also putting the CH field of their hello messages to null. This launches the clustering phases of these nodes and rearranges the cluster.

2) Modification of the TT- edge value: A cluster head including one of its neighbors has a higher value TT-edge re-initializes the CH field of its messages hello to null. When receiving updates to the hello messages, the neighbor's cluster-head that are members of the cluster propagate this re-initialization to all members of the cluster. This leads to a failure situation of the cluster-head.

5.2 Breaking Trust Relationships

When a node is authenticated as a member of the group and has at least a TT or PT relationship with a cluster member, he remains a member of the cluster. When the TT or PT relation-ships that link the node to the cluster are broken (because of mobility or change of trust value), the node must to be excluded from the cluster: it is considered a malicious node. At this level, exclusion does not have a practical impact. In fact, if the node is malicious, it does not will not respect the clustering rules. The node then joins another cluster if it has TT or PT relationships with other nodes in the network. This modification may generate other changes in the constitution of clusters. Note here that all stages of clustering are based on trust and strict respect for Clustering rules for members. Malicious nodes may not respect these rules. In this case, using the confidence values, it is possible to detect the malicious nodes. As described earlier in this section, a node can detect that a neighbor is not complying with clustering rules by controlling values of its clustering variables (contained in its hello messages).

5.3 Failure of a Core Member or a Member of the Periphery

When a kernel member or a member of the periphery fails, the other periphery members that depend on it are isolated from the cluster. These re-execute Phase 3 clustering to select a new father in the cluster or to join another cluster.

5.4 Management of New Nodes

When a new member j becomes neighbor of a member i, i assigns the value 1his trust relationship with $j(tv(i, j) =$

1). This means that a new member he is granted a TT relationship upon his arrival and, therefore, has access to the different clustering steps. However, if i and j fail to authenticate, they attribute to each other a relationship of mistrust.

6 Evaluation

We simulated our algorithm to evaluate its performance and compare it with other clustering algorithms performance. In This section provides an overview of our simulation model and results we have obtained. We simulated our approach within the platform NS2 network simulation. Our simulation MANET models a maximum of 100 nodes moving randomly in an area of 1000x1000 m2 under model Random waypoint Model [4]. Each node is equipped with a radio transceiver capable of transmitting up to 250 m.

We use as 802.11 protocols MAC layer in our experiments. We assessed the stability of our system clustering by studying the variation in the number of clusters it generates. To see the behavior of this approach and to measure the effect that will cause the implementation of our algorithm in an OLSR network, we performed several simulations with variable number of nodes and different nodes velocity. We used NS2 [7] as a network simulator.

We performed simulations with, and without clustering inter-val and we have recorded the average number of clusters built (which we note NC) and the average time during which a cluster is Maintained.

6.1 Trust Value of Cluster Head Based on the Number of Nodes

To approve the efficiency of our algorithm, we compared it with another algorithm in the literature, which is the algorithm of clustering based on node density. We notice that the trust values of the clusterhead in our proposal are much more important than in the algorithm based on density.

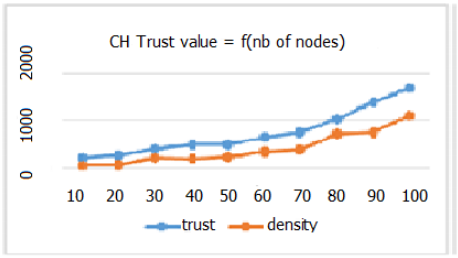


Figure 4: Average trust value of CH = f (nb of nodes), V = 10 m/s

In our algorithm the trust of the CH varies between 224,07 and 1673,9 while in the algorithm of clustering based on density it varies between76,076 and 1100, 7 (see Figure 4).

6.2 Number of Clusters Formed Based on the Number of Nodes in the Network

Figure 5 shows the evolution of the number of clusters in relation to the number of nodes in the network for a maximum speed of 10 m /s.

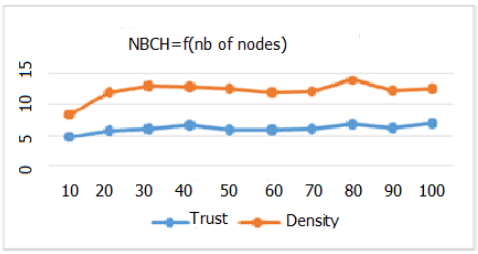


Figure 5: Average number of cluster = f (nbr nodes), V = 10m/s

We notice that the numbers of clusters in our proposal are less than in the algorithm based on density, which shows the stability of our proposal.

6.3 Number of Clusters Formed Based on the Number of Nodes in the Network

Figure 6 shows the evolution of the number of clusters in relation to the number of nodes in the network for a maximum speed of 10 m /s.

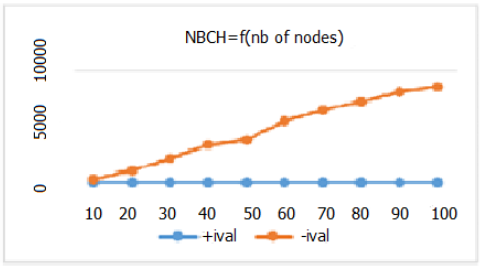


Figure 6: Average number of cluster = f (nbr nodes), V = 10m/s

We notice a great improvement with the use of the clustering interval. The number of clusters varies between 246 and 8588 in the case where the clustering interval is not used, when this number varies between 4.8 and 6.8 with the use of clustering interval for a network with 100 nodes.

6.4 Trust Value of Cluster Head Based on the Number of Nodes

Figure 7 shows the evolution of trust value of clusters in relation to the number of nodes in the network for a maximum speed of 10 m /s.

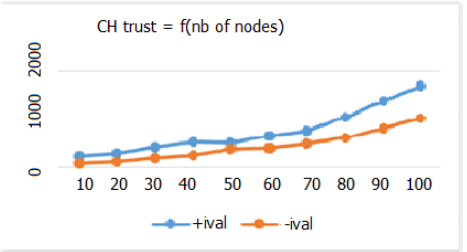


Figure 7: Average trust value of CH = f (nb of nodes), V = 10 m/s

We notice a great improvement with the use of the clustering interval. The trust value varies between 88,9 and 1112,5 in the case where the clustering interval is not used, when it varies between 224,07 and 1673.9 with the use of clustering interval for a network with 100 nodes.

6.5 Average Cluster Duration Based on the Number of Nodes in the Network

Figure 8 shows the behavior of the average time during which a cluster is built based on the number of nodes in the network.

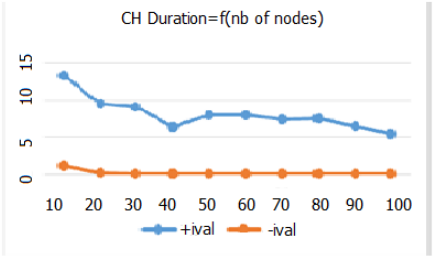


Figure 8: Average cluster duration = f(nbr nodes) , V = 10m/s

We notice a significant improvement brought by the clustering interval. The average duration of clusters varies between 0.007 ms and 1.116 ms in the case where the clustering interval is not used, when this number varies between 5,39 ms and 13.37 ms with the use of clustering interval for a network with 100 nodes.

7 Comparison and Analysis

We compared our clustering algorithm with two existing schemes SGCP [19] (Secure Group Communication Protocol)and LID [5](Lowest IDentifier). LID is one of the most known protocols clustering. LID is usually used as a reference protocol evaluation of clustering algorithms performance. We simulated the three protocols in the three scenarios described above mobility: mobility low, medium, high mobility and mobility. We considered different connectivity rate and measured the number of clusters realized by each three protocols.

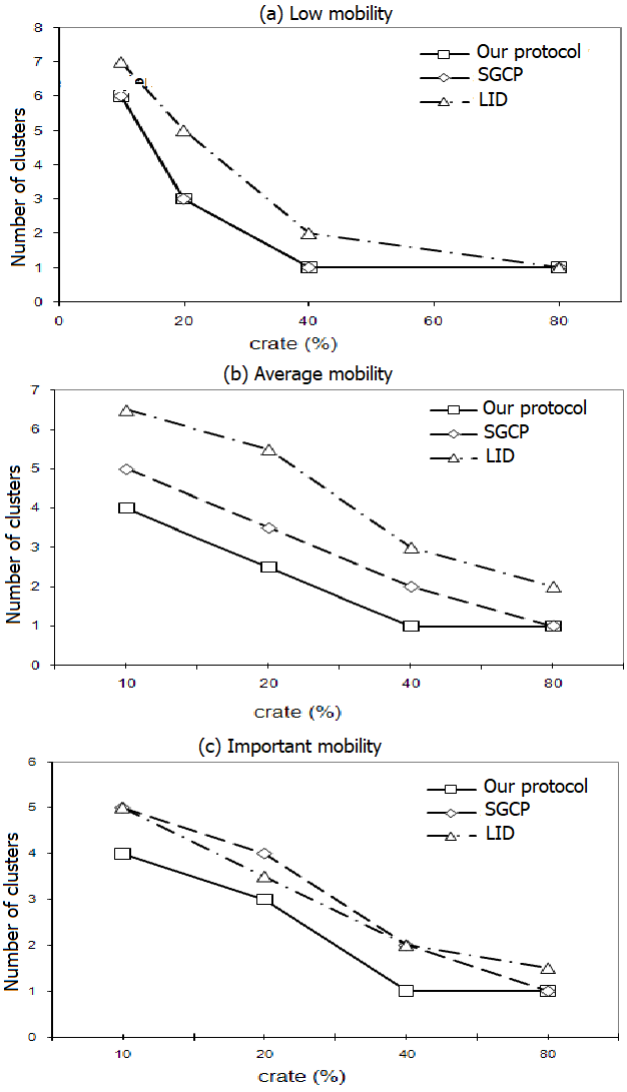


Figure 9: Comparison of the number of clusters of our protocol with LID, SGCP

Figure 9 shows that our algorithm gives good results. It has the same number of clusters as LID in a low mobility scenario(see Figure 9(a)) and has the lowest number of clusters in medium and high mobility scenarios.(see Figure 9(b),(c) We also studied the variation of the number of clusters of three protocols over time. For this, we also performed the three protocols under the same conditions of mobility and connectivity for 50s.

Figure 10 represents our results. It's clearly shows that the variation in the number of clusters is not important and is stable with LID and our protocol. It is not the case for SGCP where the number of clusters increases dramatically after 20s simulation. This shows that compared to SGCP and LID, our protocol Clustering is stable: it generates a reasonable number of clusters in all mobility scenarios.

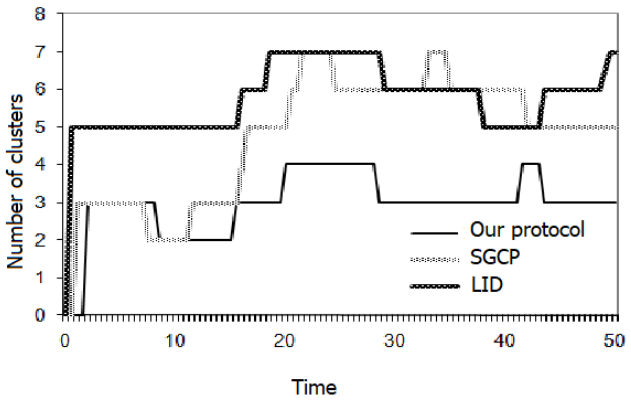


Figure 10: Évolution of the number of clusters

8 Conclusions

Our clustering algorithm is used to manage the network dynamics, it is based on building a topology to minimize the mobility of the network, optimize the scalability, facilitate and secure protocols communication. Our proposal is based on the definition of a model Dynamic and distributed trust for ad hoc mobile networks. Our approach is to divide the network into clusters organized into subtrees linked and supervised by cluster-heads. By Therefore, the addition or removal of a member only affects the cluster to which it belongs. The security of our protocol is enhanced by its clustering criterion that constantly monitors the relationship of trust between nodes and expels malicious nodes in the broadcast session.

The clustering algorithm is self-stabilizing. It runs continuously and read just clusters based on trust relationships between nodes. Relationships confidence evolves over time depending on the interactions between the nodes. The Mobility can also change the situation of clusters.

To succeed clustering, despite the presence of malicious nodes, the honest nodes cooperate closely. They do not communicate the message clustering malicious nodes and ignore all messages from clustering these nodes. Thus, clustering messages and data dissemination spend only by TT relation-ships or relationships PT. However, even if all malicious nodes were detected, clustering can be disrupted. The condition on the number of malicious nodes and their dispersion in the network is necessary. As perspective to this work to make our algorithm more stable, we added the concept of the threshold of trust, which represents the trust value at which each node can act as cluster head.

Our algorithm gives good results. It has the same number of clusters as LID in a low mobility scenario and has the lowest number of clusters in medium and high mobility scenarios. This shows that compared to SGCP and LID, our protocol Clustering is stable: it generates a reasonable number of clusters in all mobility scenarios. According to the results of simulations that we made, we notice a great improvement and better system stability with the adopted solution.Also, we plan to use the clustering solution to manage cryptographic key in MANETs.

References

[1] A. Akshai, G. Savita, C. Nirbhay, and J. Keyurbhai, "Trust based secure on demand routing protocol (tsdrp) for manets," in *Fourth International Conference on Advanced Computing & Communication Technologies (ACCT'14)*, pp. 432–438, IEEE, 2014.

[2] P. Asad Amir and M. Chris, "Establishing trust in pure ad-hoc networks," in *Proceedings of the 27th Australasian Conference on Computer Science*, vol. 26, pp. 47–54, 2004.

[3] J. H. Cho, I. R. Chen, and K. S. Chan, "Trust threshold based public key management in mobile ad hoc networks," *Ad Hoc Networks*, vol. 44, pp. 58–75, 2016.

[4] B. Christian, H. Hannes, and C. Xavier, "Stochastic properties of the random waypoint mobility model," *Wireless Networks*, vol. 10, no. 5, pp. 555–567, 2004.

[5] B. Dennis and E. Anthony, "The architectural organization of a mobile radio network via a distributed algorithm," *IEEE Transactions on communications*, vol. 29, no. 11, pp. 1694–1701, 1981.

[6] R. Eid, S. Muhammad, N. Syed Hussnain Abbas, B. Khan, and U. Kamran, "Energy efficient secure trust based clustering algorithm for mobile wireless sensor network," *Journal of Computer Networks and Communications*, vol. 2017, 2017.

[7] K. Fall, K. Varadhan, *The NS Manual*, 2003.

[8] T. George and B. John, "On trust models and trust evaluation metrics for ad hoc networks," *IEEE Journal on Selected Areas in Communications*, vol. 24, no. 2, pp. 318–328, 2006.

[9] T. George and B. John, "A testbed for comparing trust computation algorithms," in *Proceedings of the 25th Army Science Conference (ASC'06)*, 2006.

[10] K. Gomathi, B. Parvathavarthini, and C. Saravanakumar, "An efficient secure group communication in manet using fuzzy trust based clustering and hierarchical distributed group key management," *Wireless Personal Communications*, vol. 94, no. 4, pp. 2149–2162, 2017.

[11] V. S. Janani and M. S. K. Manikandan, "Mobility aware clustering scheme with bayesian-evidence trust management for public key infrastructure in ad hoc networks," *Wireless Personal Communications*, vol. 99, no. 1, pp. 371–401, 2018.

[12] P. Jay, K. Rakesh, K. Sarvesh, and J. P. Saini, "A multi-metric-based algorithm for cluster head selection in multi-hop ad hoc network," in *Next-Generation Networks*, pp. 513–524, Springer, 2018.

[13] R. H. Jhaveri, N. M. Patel, and D. C. Jinwala, "A composite trust model for secure routing in mobile ad-hoc networks," in *Ad Hoc Networks*, InTech, 2017.

[14] H. Jingsha, Z. Z. Ali, M. M. Qasim, H. M. Iftikhar, M. S. Pathan, Z. Nafei, "An efficient trust-based

scheme for secure and quality of service routing in manets," *Future Internet*, vol. 10, no. 2, p. 16, 2018.

[15] X. Li, S. Jill, and Y. Shaokai, "Evaluating trust in mobile ad hoc networks," in *The Workshop of International Conference on Computational Intelligence and Security*, Citeseer, 2005.

[16] A. Mansouri and M. S. Bouhlel, "Self-stabilizing clustering algorithm in mobile ad hoc networks," in *SAI Intelligent Systems Conference (IntelliSys'15)*, pp. 978–983, IEEE, 2015.

[17] C. Nirbhay, A. Akshai, G. Savita, and J. Keyurbhai, "Performance analysis of tsdrp and aodv routing protocol under black hole attacks in manets by varying network size," in *Fifth International Conference on Advanced Computing & Communication Technologies (ACCT'15)*, pp. 320–324, IEEE, 2015.

[18] M. Rajkumar and S. Subramanian, "A preference-based protocol for trust and head selection for cluster-based manet," *Wireless Personal Communications*, vol. 86, no. 3, pp. 1611–1627, 2016.

[19] Y. M. Tzeng, C. C. Yang, and D. R. Lin, "A secure group communication protocol for ad hoc wireless networks," *Advances in Wireless Ad Hoc and Sensor Networks, Signals and Communication Technology Series*, Springer, pp. 102–130, 2007.

[20] L. Zhaoyu, J. Anthony, and T. Robert, "A dynamic trust model for mobile ad hoc networks," in *10th IEEE International Workshop on Future Trends of Distributed Computing Systems (FTDCS'04)*, pp. 80–85, 2004.

Biography

Ali Mansouri is an assistant professor at Jendouba University since 2011; Head of Computer Department at Higher Institute of Applied Languages and Computer Sciences in Beja since 2013. Researcher at the SETIT Research Laboratory: Research Unit: Science and Technology of Image and Telecommunications (from the University of Sfax), member of PRISMa Laboratory Nautibus Building (ex 710) Claude Bernard University Lyon 1 Research field: Dynamic algorithms for ad hoc network communication, Coloration of graphs. Computer Science Master: INSA DE LYON: The National Institute of Applied Sciences of Lyon in France.Master's degree in Computer Science Management Faculty of Jendouba Tunisia.

Mohamed Salim Bouhlel is a full professor at Sfax University, Tunisia. Head of the Research Lab SETIT since 2003. President and founder of the Tunisian association on HMI since 2013. Editor in Chief of the international Journals "HMI", "MLHC and a dozen of special issues. Chairman of many international conferences research interests: Image processing, Telecom and HMI in which he has obtained more than 20 patents so far. More than 500 articles were published in IJ, IC & books.

Two Number-guessing Problems Plus Applications in Cryptography

Xingbo Wang
(Corresponding author: Xingbo Wang)
Department of Mechatronic Engineering,Foshan University
Guangdong Engineering Center of Information Security for Intelligent Manufacturing System
xbwang@fosu.edu.cn; dr.xbwang@qq.com
(Received Jan. 22, 2018; Revised and Accepted May 7, 2018; First Online Feb. 13, 2019)

Abstract

The article first puts forward two number-guessing problems that is to guess if an odd integer or one of its divisors is a divisor of another odd integer that is contained in an given odd sequence consisting in consecutive odd integers, then solves the two problems through an investigation on the global intrinsic properties of the odd sequence. Several criteria are proved to determine if a special term is contained in an odd sequence. Based on the proved criteria, algorithms are designed to detect if a interval contains a divisor-host that has a common divisor with a given number and to find out the divisor-host by means of probabilistic search. The theory and the algorithms are helpful in cracking a password or some encryption codes.

Keywords: Cryptography; Number Guessing; Password Cracking; Probabilistic Algorithm

1 Introduction

It is mandatory for a researcher of information security to guess the key of encrypted information. For example, it is necessary to simulate a crack on an encrypted system by guessing the password to see if the designed password is sure to be save enough, as J Lopez and H C Chou introduced in their articles [4] and [2]. In fact, such guessing games have been a primary approach in cryptography. In order to increase the probability of a successful encryption or decryption, kinds of mathematical methods are applied to the guessing process, as introduced and overviewed in articles [5] to [3]. Among the historical guessing approaches, the Pollard's rho method of factoring integers was a remarkably successful one because it was essentially a probabilistic algorithm called Monte Carol's approach, as analyzed in Bach's article [1].

A recent study comes across a problem that requires knowing an odd integer N with a nontrivial divisor that can divide another odd integer o that is hidden in a very large odd interval I_{odd}, which consists in finite consecutive odd integers. For example, suppose N is an odd composite integer; according to theorems proved in [7] and [8], one of N's divisor p can be found uniquely in an odd integer o lying in a *divisor-interval*, as the five odd composite integers and their respective divisor-intervals list in Table 1.

Due to a vast number of the odd integers contained in I_{odd}, a one-by-one sequential check is impossible to perform the detection of o, called a *divisor-host*, even with the fastest computer in the world. Hence the interval I_{odd} is subdivided into finite subintervals to be separately searched in parallel computation, as introduced in article [8]. Since there might be only one divisor-host o contained in one of the subdivided subintervals, solutions for the following two problems are necessary.

(P1). Is there a way to know which subinterval o lies in?

(P2). Is there a simple and fast way to locate o's position if the subinterval in which o lies is known?

The problems P1 and P2 are sure the kind of guessing-number problems if their background stated above is ignored. Moreover, they can be described in the following more general questions Q1 and Q2.

(Q1). Is there a way to know if an odd integer o itself or one of its divisorscan divide one of the odd integers in an odd interval I_{odd}?

(Q2). If an odd integer o itself or one of its divisors does divide one of the odd integers in an odd interval I_{odd}, is there a simple and fast way to detect the *divisor-host* that is divisible by o or one of o's divisors?

This article intends to answer the two questions Q1 and Q2 and explores their applications. By analyzing the intrinsic traits of consecutive odd integers that contain a special term, the article proves several criteria that can determine if o or one of o's divisors is contained in an odd interval or in an arithmetic progression. Based on the proved criteria, a fast way is developed to detect the

Table 1: Big integers and their divisor-intervals

Composite integer N	N**'s Divisor-interval**
$N_1 = 1123877887715932507$	[323935746324954482071652928163131137,323935746324954482071652929223262207]
$N_2 = 1129367102454866881$	[325517904753935056870231790741633615,325517904753935056870231791804350463]
$N_3 = 35249679931198483$	[317500890806149478235110120566155,317500890806149478235110308315135]
$N_4 = 208127655734009353$	[1499717812494687035718622130777530 9,1499717812494687035718622176398540 7]
$N_5 = 331432537700013787$	[477644625050956786725043756492054 87,477644625050956786725043762249072 63]

interval that contains o or one of o's divisors and a probabilistic approach is proposed to find out the integer that has a common divisor with o.

2 Symbols and Notations

In this whole article, an odd interval $[a, b]$, also called an odd sequence, is a set of consecutive odd numbers that take a as the lower bound and b as the upper bound, for example, $[3, 11] = \{3, 5, 7, 9, 11\}$. Symbol $\Sigma[a, b]$ is to express the sum of all terms on the close odd interval $[a, b]$, for example, $\Sigma[3, 11] = 3 + 5 + 7 + 9 + 11 = 35$. Symbol $\frac{\Sigma S}{n}$ is the arithmetic average value on odd interval S that consists in n odd integers, and symbol $\frac{\Sigma S}{\rho n} = \frac{1}{\rho} \times \frac{\Sigma S}{n}$ is called a ρ-enhanced arithmetic average value on S. Symbol $]a, b[$ is to express a set whose elements are either smaller than a or bigger than b, and thus $x \in]a, b[$ is equivalent to $x \leq a$ or $x \geq b$. Symbol $a|b$ means b is divisible by a and symbol $a \nmid b$ means b is not divisible by a. Symbol $A \Rightarrow B$ means conclusion B can be derived from condition A; $A \Leftrightarrow B$ means B holds if and only if A holds; the term 'if and only if' is also briefly written by notation *iff*, which also means 'the necessary and sufficient condition.

3 Main Results and Proofs

Theorem 1. *Suppose N is a composite odd integer with a divisor p bigger than 1, m and n are positive integers with $1 \leq m \leq n < p$; let $S = \{s_1, s_2, ..., s_n\}$ be a set that consists in n consecutive odd integers in which s_m is the unique term such that $p|s_m$ and $N \nmid s_m$; then there are at least $(n-m)(n-m+1)+1$ possible ways to compute $GCD(N, s_m)$ if $m \leq n < 2m-1$, and there are at least $m(m-1)+1$ possible ways to compute $GCD(N, s_m)$ if $n \geq 2m-1$.*

Proof. The given conditions immediately yield $GCD(s_m, N) = p$. Since S consists in n consecutive odd integers, when $n \geq 2m-1$ it knows that, there are $m-1$ terms on the left side of s_m and there are at least $m-1$ terms on its right side. This time, it yields the following facts.

1) There are $2(m-1)$ ways to obtain $2s_m$ by choosing s_{m-j} or s_{m+j} that fits $s_{m-j} + s_{m+j} = 2s_m$ with $j = 1, 2, ..., m-1$;

2) There are $2(m-2)$ ways to obtain $4s_m$ by choosing two consecutive terms, s_{m-j} and s_{m-j-1}, or two terms, s_{m+j} and s_{m+j+1}, that fit $s_{m-j} + s_{m-j-1} + s_{m+j} + s_{m+j+1} = 4s_m$ with $j = 1, 3, ..., m-2$. Considering that arbitrary symmetrically-distributed four terms, say s_{m-l}, s_{m-k},s_{m+l} and s_{m+k}, fit $s_{m-l} + s_{m-k} + s_{m+l} + s_{m+k} = 4s_m$, one knows there are at least $2(m-2)$ ways to obtain $4s_m$.

3) Likewise, there are at least $2(m-k)$ ways to obtain by choosing k consecutive terms, say s_{m-j-k}, $\cdots$, s_{m-j-1}, s_{m-j} or s_{m+j}, s_{m+j+1}, $\cdots$, s_{m+j+k} that fit $s_{m-j-k} + \cdots + s_{m-j-1} + s_{m-j} + s_{m+j} + s_{m+j+1} + \cdots + s_{m+j+k} = 2ks_m$ with $j = 1, \cdots, m-k-1$;

4) There is one way to obtain s_m, that is, to choose s_m itself.

Consequently, from 1 to k the minimal ways, denoted by Λ_k, which can produce multiples of s_m are given by

$$\begin{aligned}\Lambda_k &= 2(m-1) + 2(m-2) + \cdots + 2(m-k) + 1 \\ &= k(2m-k-1) + 1.\end{aligned}$$

And when $k = m-1$, it yields

$$\Lambda_{m-1} = m(m-1) + 1.$$

When $m \leq n < 2m-1$, there are $m-1$ terms on the left side of s_m while there are merely $n-m < m-1$ terms on its right side. Hence Λ_{n-m} is the biggest number for the case and consequently it yields

$$\begin{aligned}\Lambda_{n-m} &= 2(n-m) + ... + 2 + 1 \\ &= 2(\frac{(n-m+1)(n-m)}{2}) + 1 \\ &= (n-m)(n-m+1) + 1\end{aligned}$$

□

Corollary 1. *Suppose p is an odd prime number, m and n are positive integers with $1 \leq m < n < p$ and $n = 2m-1$; let $S = \{s_1, s_2, ..., s_n\}$ be a set consisting in n consecutive odd integers; then the necessary and sufficient condition of $p|s_m$ is $p|\Sigma S$, namely, $p|s_m \Leftrightarrow p|\Sigma S$.*

Proof. $n = 2m-1$ means s_m is the mid-term of S. $p|s_m$ yields $s_m = ps$ for some odd integer $s > 1$ and $s_{m-j} + s_{m+j} = 2s_m$ with $j = 1, 2, ..., m-1$. Consequently,

$$p|s_m \Rightarrow p|\Sigma S.$$

Since S consists in n consecutive odd integers, it knows

$$\begin{aligned}\Sigma S &= \Sigma[s_1, s_{m-1}] + s_m + \Sigma[s_{m+1}, s_{2m-1}] \\ &= (2m-1)s_m.\end{aligned} \tag{1}$$

Note that the condition that $1 \le m < n < p$ and $n = 2m-1$ yields $p > 2m-1$; hence $p \nmid 2m-1$ because of p's primality and as a result,

$$p|\Sigma S \Rightarrow p|s_m.$$

□

Corollary 2. *Suppose p is an odd prime number, m and n are positive integers with $1 \le m < n < p$ and $n = 2m-1$; Let $S = \{s_1, s_2, ..., s_n\}$ be an arithmetic integer sequence that consists in n consecutive terms; then the necessary and sufficient condition of $p|s_m$ is $p|\Sigma S$, namely, $p|s_m \Leftrightarrow p|\Sigma S$.*

Proof. (Omitted) □

Theorem 2. *Let S be an odd sequence that consists in n consecutive odd integers and s be an odd integer; then S contains s iff $ns-n(n-1) \le \Sigma S \le ns+n(n-1)$, whereas S does not contain s iff $\Sigma S \in]ns-n(n+1), ns+n(n+1)[$.*

Proof. The necessity. When containing s, S is given by

$$S = \{\underbrace{s-2k, ..., s-2, s, s+2, ..., s+2l}_{n \quad terms}\}$$

with $l, k \ge 0$ and $l+k+1 = n$; consequently it yields

$$\Sigma S = ns + (l-k)n.$$

Since $k+l+1 = n \Rightarrow k+l = n-1 \Rightarrow \begin{cases} l = n-1-k \\ k = n-1-l \end{cases}$, it yields

$$\Sigma S = \begin{cases} ns + n(n-1) - 2nk \\ ns - n(n-1) + 2nl \end{cases} \tag{2}$$

Considering the following two particular cases given by

$$S = \{\underbrace{s, s+2, ..., s+2(n-1)}_{n \quad terms}\} \Rightarrow \Sigma S = ns + n(n-1)$$

and

$$S = \{\underbrace{s-2(n-1), ..., s-2, s}_{n \quad terms}\} \Rightarrow \Sigma S = ns - n(n-1),$$

it yields

$$ns - n(n-1) \le \Sigma S \le ns + n(n-1).$$

When not containing s, S fits one of the following two cases

1) $S = \{\underbrace{s+2k, s+2k+2, ..., s+2k+2(n-1)}_{n \quad terms}\}$ with $k \ge 1$ leads to

$$\Sigma S = ns + n(n-1) + 2kn \tag{3}$$

2) $S = \{\underbrace{s-2l-2(n-1), ..., s-2l-2, s-2l}_{n \quad terms}\}$ with $l \ge 1$ leads to

$$\Sigma S = ns - n(n-1) - 2ln. \tag{4}$$

Note that, Case (1) turns to be $S = \{\underbrace{s+2, s+4, ..., s+2n}_{n \quad terms}\} \Rightarrow \Sigma S = ns+n(n+1)$ when $k = 1$ and case (2) turns to be $S = \{\underbrace{s-2n, ..., s-4, s-2}_{n \quad terms}\} \Rightarrow \Sigma S = ns - n(n+1)$ when $l = 1$; it immediately knows $\Sigma S \in]ns - n(n+1), ns + n(n+1)$.

Hence the necessity is true. Next is the proof for the sufficiency. Without loss of generality, let $S = \{s_1, s_2, ..., s_n\}$; then $s_n = s_1 + 2(n-1)$ and $\Sigma S = ns_1 + n(n-1)$. The condition $ns-n(n-1) \le \Sigma S \le ns+n(n-1)$ yields

$$\frac{\Sigma S}{n} - (n-1) \le s \le \frac{\Sigma S}{n} + (n-1).$$

Namely,

$$s_1 \le s \le s_1 + 2(n-1).$$

Hence it is sure that S contains s.

Now consider $\Sigma S \in]ns-n(n+1), ns+n(n+1)[$, namely, $\Sigma S \le ns - n(n+1)$ or $\Sigma S \ge ns + n(n+1)$. If $\Sigma S \le ns - n(n+1)$, it holds

$$\frac{\Sigma S}{n} + (n+1) \le s.$$

Substituting ΣS by $ns_1 + n(n-1)$ yields, namely

$$s_n = s_1 + 2(n-1) < s$$

which says S does not contains s when $\Sigma S \le ns-n(n+1)$.

Similarly, $\Sigma S \ge ns+n(n+1)$ yields $s < s_n - 2(n-1) = s_1$, which indicates S does not contains s.

Hence the sufficiency is also true. □

Theorem 2 can be alternatively stated as the following Theorem 3 and Theorem 4.

Theorem 3. *Let S be an odd sequence that consists in n consecutive odd integers and s be an odd integer; then S contains s iff $|\frac{\Sigma S}{n} - s| \le n-1$, and S does not contain s iff $|\frac{\Sigma S}{n} - s| \ge n+1$.*

Theorem 4. *Let S be an odd sequence that consists in n consecutive odd integers and s be an odd integer; then S contains s iff $|\frac{\Sigma S}{ns} - 1| \le \frac{n-1}{s}$, and S does not contain s iff $|\frac{\Sigma S}{ns} - 1| \ge \frac{n+1}{s}$.*

These theorems directly derive out the following corollaries.

Corollary 3. *Let S be an arithmetic integer sequence that consists in n terms with common difference d; suppose s is an integer number; if S contains s, then $ns-\frac{1}{2}n(n-1)d \leq \Sigma S \leq ns+\frac{1}{2}n(n-1)d$; if S does not contains s, then $\Sigma S \in]ns-\frac{1}{2}n(n+1)d, ns+\frac{1}{2}n(n+1)d[$.*

Proof of Corollary 3. (Omitted) □

Corollary 4. *Let $S_1 = \{s_1, s_2, ..., s_n\}$ and $S_2 = \{s_n + 2, s_n + 4, ..., s_n + 2n\}$ be two adjacent odd sequences each of which consists in n terms; then $\Sigma S_2 = \Sigma S_1 + 2n^2$.*

Proof. $s_n = s_1 + 2(n-1) \Rightarrow \Sigma S_1 = ns_1 + n(n-1)$. $\Sigma S_2 = ns_n + n(n+1) = ns_1 + n(n-1) + 2n^2 = \Sigma S_1 + 2n^2$. □

Corollary 5. *Suppose S is an odd sequence that consists in n consecutive odd integers and $s = pq$ is an integer with p and q being odd integers and $1 < n \leq p \leq q$; then S contains s iff $|\frac{\Sigma S}{np} - q| \leq \frac{n-1}{p}$, and S does not contain s iff $|\frac{\Sigma S}{np} - q| \geq \frac{n+1}{p}$. This indicates $\frac{\Sigma S}{np}$ is more closer to q if S contains $s = pq$.*

Proof. (Omitted) □

Corollary 6. *Suppose S is an odd sequence that consists in n consecutive odd integers and p is an odd integer that fits $1 < n \leq p$; if there exists an odd integer q that satisfies $|q - \frac{\Sigma S}{np}| \leq \frac{n-1}{p}$ then S contain $s = pq$.*

Proof. $|q - \frac{\Sigma S}{np}| \leq \frac{n-1}{p} \Rightarrow -\frac{n-1}{p} \leq q - \frac{\Sigma S}{np} \leq \frac{n-1}{p} \Rightarrow \frac{\Sigma S}{np} - \frac{n-1}{p} \leq q \leq \frac{\Sigma S}{np} + \frac{n-1}{p} \Rightarrow \Sigma S - n(n-1) \leq ns \leq \Sigma S + n(n-1)$. Since p and q are odd integers, $s = pq$ is an odd integer. By Theorem 2, it knows S contain $s = pq$. □

Corollary 5 and Corollary 6 result in the following Theorem 5.

Theorem 5. *Suppose S is an odd sequence that consists in n consecutive odd integers and p is an odd integer that fit $1 < n \leq p$; then there exists an odd integer q such that S contains $s = pq$ iff there exists an odd integer q that satisfies $|q - \frac{\Sigma S}{np}| \leq \frac{n-1}{p}$.*

Obviously, the computation of q is mandatory to take into consideration, as proposed by the following proposition.

Proposition 1. *Arbitrary integers p, q and n with $1 < n \leq p$ and $q \geq 1$, the inequality $|q - \frac{\alpha}{np}| \leq \frac{n-1}{p}$ yields with arbitrary real number α. Consequently, suppose S is an odd sequence consisting in n consecutive odd integers and p is an odd integer that fit $1 < n \leq p$; if there exists an odd integer q such that S contains $s = pq$ then $\left\lfloor \frac{\Sigma S}{np} \right\rfloor - 1 \leq q \leq \left\lfloor \frac{\Sigma S}{np} \right\rfloor + 1$.*

Proof. $\frac{\alpha}{np} - \frac{n-1}{p} \leq q \leq \frac{\alpha}{np} + \frac{n-1}{p} \Rightarrow \left\lfloor \frac{\alpha}{np} - \frac{n-1}{p} \right\rfloor \leq q \leq \left\lfloor \frac{\alpha}{np} + \frac{n-1}{p} \right\rfloor \Rightarrow \left\lfloor \frac{\alpha}{np} \right\rfloor - \left\lfloor \frac{n-1}{p} \right\rfloor - 1 \leq q \leq \left\lfloor \frac{\alpha}{np} \right\rfloor + \left\lfloor \frac{n-1}{p} \right\rfloor + 1$. Since $1 < n \leq p$ leads to $\left\lfloor \frac{n-1}{p} \right\rfloor = 0$, it knows

$$\left\lfloor \frac{\alpha}{np} \right\rfloor - 1 \leq q \leq \left\lfloor \frac{\alpha}{np} \right\rfloor + 1$$

□

4 Algorithms & Applications

By now, the questions Q1 and Q2 raised in the introductory part has been answered. For example, Theorem 2 shows how to know if an odd interval contains a special odd integer, Theorem 5 shows how to know if a divisor of an odd integer is a divisor of a term in an odd interval. Moreover, Theorem 1 indicates that, there is big probability to seek a unique term in a large odd interval, This provides a new applicable chance in factorization of odd composite integers by probabilistic approaches. Accordingly, this section presents a fast way to detect the divisor-interval that contains the divisor-host and introduces a probabilistic approach to search the divisor-host. Examples to solve the problems Q1 and Q2 are also given in this section.

4.1 Fast Detection of Divisor-interval

Let I_{odd} be a very large odd interval that contains the divisor host N_{host}; as is stated, a one-by-one check is difficult to locate N_{host}. Instead, a subdivision-based approach can reach an appreciated effect. The approach first subdivides I_{odd} into a proper number of subintervals, then calculates the arithmetic average value $\frac{\Sigma S}{n}$, s-enhanced arithmetic average value $\frac{\Sigma S}{ns}$ or p-enhanced arithmetic average value $\frac{\Sigma S}{np}$ on each subinterval. Since $n-1, \frac{n-1}{s}$ and $\frac{p-1}{p}$ are definitely known for a given n, s or p, it is easy to judge which subinterval contains N_{host}. Assuming L_{si} is set to be the biggest number of odd integers contained in each subinterval, the following algorithm can find the subinterval containing N_{host}. The process is described as a divisor-interval detecting algorithm (Algorithm 1).

Algorithm 1 Divisor-interval Detecting Algorithm

1: Begin
2: Input: Large odd interval I_{odd}, L_{si}, s (or p);
3: Step 1. Calculate N, the number of odd integers in I_{odd};
4: Step 2. Subdivide I_{odd} into $m+1$ subintervals, among which m ones are of equal length L_{si} and one is of length R by $N = mL_{si} + R$.
5: Step 3. Compare on each subinterval $\frac{\Sigma S}{L_{si}} - s$ with $n-1$ or $\frac{\Sigma S}{sL_{si}} - 1$ with $\frac{L_{si}-1}{s}$ for objective s, (or compute q with Algorithm 2 and then compare $\frac{\Sigma S}{pL_{si}} - q$ with $\frac{L_{si}-1}{p}$ for objective p), where ΣS is the sum on the respective subinterval.
6: Step 4. Choose the host-interval by Theorem 5* (or Corollary 5 for p).
7: End

Remark 1. *If the calculated objective is a divisor p, it is mandatory to calculate q first. This can be done as the following subroutine (Algorithm 2) shows.*

Algorithm 2 Subroutine: q's Calculation

1: Begin
2: Input: p, L_{si}, ΣS;
3: Step 1. Calculate $\varepsilon = \frac{L_{si}-1}{p}$;
4: Step 2. Calculate $q_i^{-1} = \left\lfloor \frac{\Sigma S}{pL_{si}} \right\rfloor - 1, q_i^0 = \left\lfloor \frac{\Sigma S}{pL_{si}} \right\rfloor$ and $q_i^{+1} = \left\lfloor \frac{\Sigma S}{pL_{si}} \right\rfloor + 1$ on each subinterval;
5: Step 3. Choose q to be an odd one from q_i^{-1}, q_i^0 and q_i^{+1} that satisfies $|q - \frac{\Sigma S}{np}| \le \varepsilon$
6: End

4.2 Probabilistic Approach To Find Out Divisor-host

Now that a divisor-interval is obtained with Algorithm 1, one can search the divisor-host by *the brutal search*, which searches the objective number one by one in the divisor-interval. It is known that, the efficiency of the brutal search depends on the length of the divisor-interval and sometimes a probabilistic search is faster than the brutal search. Therefore here introduces a probabilistic approach. According to Theorem 3, the probability to find the GCD will increase a lot by adding two or more odd integers in the divisor-interval. Thus, the probabilistic approach of searching the divisor-host can be deigned as Algorithm 3 shows.

Algorithm 3 Probabilistic Algorithm

1: Begin
2: Input: odd composite integer N and N's divisor-interval I_{odd}
3: Begin loop
4: Step 1. Select an integer $a \in I_{odd}$ randomly;
Select another integer $b \in I_{odd}$ randomly;
5: Step 2. d_1=FindGCD(N, a);d_2=FindGCD(N, b);
d_3=FindGCD(N, $a+b$);
6: Step 3. If $d_1 > 1$ then stop loop and return d_1;
If $d_2 > 1$ then stop loop and return d_2;
If $d_3 > 1$ then stop loop and return d_3;
7: End loop
8: End

Remark 2. *Algorithm 3 can also be a algorithm to factorize an odd integer. It can be applied only on the divisor-interval. Otherwise, it will fail absolutely.*

Applying Algorithm 3, big odd integers list in Table 1 are very quickly factorized by $N_1 = 299155897 \times 3756830131$, $N_2 = 25869889 \times 43655660929$, $N_3 = 59138501 \times 596052983$, $N_4 = 430470917 \times 483488309$ and $N_5 = 114098219 \times 2904800273$.

4.3 Comments & Examples

One can see that, the purpose of the algorithm 1 is to subdivide a big interval to find out the host-interval. The time complexity of the algorithm is $O(\frac{N}{L_{si}})$ with N being the number of terms in I_{obj} and L_{si} that is set up in advance and according to the computational capability of the computer performing the computation. Algorithm 2 is an auxiliary process of Algorithm 1 and its time complexity is $O(1)$. Algorithm 3 can be applied following Algorithm 1. It needs to point out that, Algorithm 3 here is merely a framework of the probabilistic approach. It needs improving in the way that picks the numbers randomly. The related work will be shown in a future paper.

In order for readers to understand the theory and the algorithms 1 and 2, here presents 2 examples which are also examples of the problems Q1 and Q2 with their solutions.

Example 1. *Let S_1, S_2 and S_3 be three sets each of which consists in 5 odd consecutive integers; suppose the sum of all the odd integers in each set is given by 4045, 4095 and 4145 respectively, judge which of S_1, S_2 and S_3 contains the number 825.*

Solution. The number *825* is the objective and each of S_1, S_2 and S_3 is a possible host-interval. $n = 5$ yields $n-1 = 4$. The arithmetic average values of S_1, S_2 and S_3 are respectively $\frac{4045}{5} = 809$, $\frac{4095}{5} = 819$ and $\frac{4145}{5} = 829$. Since $|809 - 825| = 16$, $|819 - 825| = 6$ and $|829 - 825| = 4$, it knows S_3 contains the objective *825* by Theorem 5*.

Example 2. *Let S={1133, 1135, 1137, 1139, 1141, 1143, 1145, 1147, 1149, 1151, 1153, 1155, 1157, 1159, 1161, 1163, 1165, 1167, 1169, 1171}; detect if 31 can be divisible one of the terms in S.*

Solution. The objective is $p = 31$ and $I_{obj} = S$. $N = 20$ is the number of terms in S. Subdivide S into 4 sub-sequences, (and thus $L_{si} = 5$), by

$$S_1 = \{1133, 1135, 1137, 1139, 1141\},$$

$$S_2 = \{1143, 1145, 1147, 1149, 1151\},$$

$$S_3 = \{1153, 1155, 1157, 1159, 1161\},$$

$$S_4 = \{1163, 1165, 1167, 1169, 1171\}.$$

Let $\varepsilon = \frac{L_{si}-1}{p} = \frac{5-1}{31} \approx 0.129$ and $E = \frac{L_{si}+1}{p} = \frac{5+1}{31} \approx 0.1935$; by Corollary 4, it needs to determine a q and check $|\frac{\Sigma S_i}{5\times 31} - q| \le \varepsilon$ for $i = 1, 2, 3, 4$. Note that,

$$\Sigma S_1 = 5685 \Rightarrow \frac{\Sigma S_1}{5 \times 31} \approx 36.6774$$

$$\Sigma S_2 = 5735 \Rightarrow \frac{\Sigma S_2}{5 \times 31} = 37.0000$$

$$\Sigma S_3 = 5785 \Rightarrow \frac{\Sigma S_3}{5 \times 31} \approx 37.3226$$

$$\Sigma S_4 = 5835 \Rightarrow \frac{\Sigma S_4}{5 \times 31} \approx 37.6452$$

By Proposition 1, q is 37.0000; then by Corollary 4, it knows that S_2 contains a term that is divisible by 31. In fact, 1147=31×37.

As a comparison, subdividing S into 5 sub-sequences, which leads to $L_{si} = 4$, results in

$$S_1 = \{1133, 1135, 1137, 1139\},$$

$$S_2 = \{1141, 1143, 1145, 1147\},$$

$$S_3 = \{1149, 1151, 1153, 1155\},$$

$$S_4 = \{1157, 1159, 1161, 1163\},$$

$$S_5 = \{1165, 1167, 1169, 1171\},$$

Referring to Corollary 4, let $\varepsilon = \frac{L_{si}-1}{p} = \frac{4-1}{31} \approx 0.0968$ and $E = \frac{L_{si}+1}{p} = \frac{5+1}{31} \approx 0.1613$; then

$$\Sigma S_1 = 4544 \Rightarrow \frac{\Sigma S_1}{4 \times 31} \approx 36.6452$$

$$\Sigma S_2 = 4576 \Rightarrow \frac{\Sigma S_2}{4 \times 31} \approx 36.9032$$

$$\Sigma S_3 = 4608 \Rightarrow \frac{\Sigma S_3}{4 \times 31} \approx 37.1613$$

$$\Sigma S_4 = 4640 \Rightarrow \frac{\Sigma S_4}{4 \times 31} \approx 37.4194$$

$$\Sigma S_5 = 4672 \Rightarrow \frac{\Sigma S_5}{4 \times 31} \approx 37.6774$$

Obviously, by Proposition 1, $q = 36 + 1 = 37$ is a reasonable choice and this time S_2 contains $s = 31 \times 37$ because $|\frac{\Sigma S_2}{4\times 31} - q| \approx 0.09677 \leq \varepsilon$ while $|\frac{\Sigma S_3}{4\times 31} - q| \approx 0.1613 \approx E$ and $|\frac{\Sigma S_1}{4\times 31} - q| \approx 0.3548 > E$.

5 Conclusions and Future Work

Finding a hidden objective in a set is meaningful in both mathematics and engineering. The set of consecutive odd integers, as a special set in cryptography, hides many problems inside. It is worthy of a subtitle study of the set. This paper solves the problem to detect if a number itself or one of its divisors is a divisor of another number that hidden in a set. It is helpful in cracking a password or some encryption codes. The probabilistic approach raised in this paper is sure a new idea to factorize odd integers because its way of finding GCD by adding randomly-picked items is quite different from the present Pollard's rho method that finds GCD by subtracting randomly-picked items. However, as stated before, the algorithm presented in this paper needs a further investigation. This is part of our future work. Hope it is concerned by more colleagues and valuable results come into being.

Acknowledgments

The research work is supported by the State Key Laboratory of Mathematical Engineering and Advanced Computing under Open Project Program No.2017A01, Department of Guangdong Science and Technology under projects 2015A030401105 and 2015A010104011, Foshan Bureau of Science and Technology under projects 2016AG100311, Projects 2014SFKC30 and 2014QTLXXM42 from Guangdong Education Department. The author sincerely presents thanks to them all.

References

[1] E. Bach, "Toward a theory of Pollard's Rho method", *Information and Computation*, vol. 90, pp. 139-155, 1991.

[2] H. C. Chou, H. C. Lee, H. J. Yu, *et al.*, "Password cracking based on learned patterns from disclosed passwords", *International Journal of Innovative Computing Information & Control*, vol. 9, no. 2, pp. 821-839, 2013.

[3] S. Houshmand, S. Aggarwal, R. Flood, "Next gen PCFG password cracking", *IEEE Transactions on Information Forensics & Security*, vol. 10, no. 8, pp. 1776-1791, 2017.

[4] J. Lopez, L. F. Cranor, N. Christin, *et al.*, "Guess again (and again and again): Measuring password strength by simulating password-cracking algorithms", *Security & Privacy*, vol. 12, no. 02, pp. 523-537, 2012.

[5] S. Marechal, "Advances in password cracking", *Journal in Computer Virology*, no. 4, pp. 73-81, 2008. DOI10.1007/s11416-007-0064-y

[6] V. Vijayan, J. P. Joy, M. S. Suchithra, "A review on password cracking strategies", *International Journal of Computer and Communication Technology*, vol. 3, no. 3, pp. 8-15, 2014.

[7] X. Wang, "Genetic traits of odd numbers with applications in factorization of integers", *Global Journal of Pure and Applied Mathematics*, vol. 13, no. 1, pp. 318-333, 2017.

[8] X. Wang, "Strategy for algorithm design in factoring RSA numbers", *IOSR Journal of Computer Engineering*, vol. 19, no. 3(ver.2), pp. 1-7, 2017.

[9] C. M. Weir, "Using probabilistic techniques to aid in password cracking attacks", *PhD Dissertations*, 2010. (`http://purl.flvc.org/fsu/fd/FSU_migr_etd-1213`)

[10] F. Yu, Y. Huang, "An overview of study of passowrd cracking", in *International Conference on Computer Science and Mechanical Automation*, pp. 25-29, 2015.

Biography

Dr. & Prof. Xingbo WANG was born in Hubei, China. He got his Master and Doctor degrees at National University of Defense Technology of China and had been a staff

in charge of researching and developing CAD/CAM/NC technologies in the university. Since 2010, he has been a professor in Foshan University with research interests in computer application and information security. He is now the chief of Guangdong engineering center of information security for intelligent manufacturing system. Prof. WANG was in charge of more than 40 projects including projects from the National Science Foundation Committee, published 8 books and over 90 papers related with mathematics, computer science and mechatronic engineering, and invented 30 more patents in the related fields.

General Model for Secure Electronic Cash Scheme

Dany Eka Saputra[1], Sarwono Sutikno[2], and Suhono Harso Supangkat[2]
(Corresponding author: Dany Eka Saputra)
Departement on Informatics, STMIK "AMIKBANDUNG"[1]
Jl. Jakarta no 28, Kota Bandung, Jawa Barat, Indonesia
School of Electrical Engineering, Institut Teknologi Bandung[2]
Jl. Ganesha No 10, Kota Bandung, Jawa Barat, Indonesia
(Email: dekastra@gmail.com)
(Received Nov. 01, 2017; Revised and Accepted June 18, 2018; First Online Feb. 24, 2019)

Abstract

The vast variation of electronic cash scheme make it difficult to compare a scheme with another. We propose a general model of electronic cash. The development of the model uses logical modeling based on Inenaga *et al.* model of money system. The model consists of three submodel: the model of system, the model of process, and the model of property. The model of system describe the basic model of electronic cash data, entity, and form. The model of process enlist the common process found in electronic cash scheme. The model of property describes the common property in electronic cash scheme, including security property. We find that our model can be used as a base of security evaluation and covers wider variation of electronic cash scheme than the model in Inenaga *et al.*

Keywords: Electronic Cash; Mathematic Modeling; Security Model

1 Introduction

Comparing and choosing electronic cash schemes for an implementation requires great effort. Each scheme need to be reviewed to list its respective properties and choose the one that suits the implementation. However, two different schemes may define a single property differently. This condition may complicate the effort to define the security objectives of electronic cash and may hamper the security of the implementation. For example, Dreier *et al.* proposed a method to evaluate the security of electronic cash [9]. The works of Chen & Chou [7], and Wang *et al.* [28] also attempt to evaluate another existing schemes. However, the definition of forgery in [9] differs with the definition used in [7] and [28].

An attempt to design new electronic cash scheme face similar problem. Different definition of property and behavior may lead to incorrect design. This condition may result in insecure scheme. Having standard definition or model of electronic cash and its property (including security property) greatly help the process of designing a new scheme or comparing schemes for implementation.

Modeling a common definition of electronic cash demands quite an effort. A model usually represents certain aspect of an object. By observing the object for any pattern that represent the object's properties, we can build a model of the object. In electronic cash, extracting those pattern is a challenging task. Due to the numerous scheme of electronic cash, we can find many variation of property and its definition. Under this condition, it is difficult to extract a common pattern. For a start, the electronic cash scheme can be divided into two paradigms: centralized and distributed electronic cash [26]. Both paradigms have different ways to describe and process electronic cash. Cannard-Gouget [2] explain four definition of anonymity, which is differs from the definition of anonymity in most of electronic cash scheme.

Inenaga *et al.* propose a model of money system [15]. The model describes the general model of money system that can be used to model electronic cash. The model also describes the transfer model of electronic cash and the security property of electronic cash. However, the model only covers off-line electronic scheme. It does not model the electronic cash data creation process and does not consider the issuer of electronic cash as part of the system. The model lacks the generality needed to describe electronic cash system.

We propose a new General Model of Electronic Cash. We take several concepts from Inenaga *et al.* and form a new model that covers wider range of electronic cash scheme. The model is build by using logical approach, so it can be used as a tool to design new scheme or to compare and evaluate existing scheme.

The proposed model only covers the description of centralized electronic cash. By using this paradigm, we refer electronic cash as a digital representative of cash that cre-

ated after exchanging a certain amount of cash to the electronic cash issuer. Distributed electronic cash (or cryptocurrency), such as Bitcoin [22], can not be described by using our model. The hybrid (centralized-distributed) electronic cash scheme ([8, 14, 20, 27]) also cannot be described using our model.

The rest of this paper is arranged as follows. Next, we will describe the definition of electronic cash data and system. The third part of this paper explains the model of process in electronic cash. The next part contains the property of electronic cash, including security property. We give a case study where we use our model to analyze the security of Chaum's Untraceable Electronic Cash scheme [6]. We also compare our model with Inenaga *et al.* model at the end of this paper.

2 Model of Electronic Cash System

2.1 Electronic Cash Data

The model starts with the definition of electronic cash data. The definition acts as the basis of the rest of the model. The definition of electronic cash data is as follows.

Definition 1. *Let m be a medium to store monetary value, v is a non-negative integer represents denomination of monetary value, and u is the owner's identity of a money. An electronic cash data e is defined as a function of m, v, and u, such as:*

$$e = f(m, v, u).$$

In [15], a medium is mapped to a value and the holder/owner of the electronic cash. The value function (vf) is a function that maps a medium to a value. The holder function (hf) is a mapping of a medium to a holder. We take this concept from [15] and redefine the function as follows.

Definition 2. *Given a certain medium m and a single denomination value v, where $v \in \mathbb{Z} \wedge v > 0$, a value function is defined as a function of m and v, declared as*

$$vf_m^v = f(m, v).$$

The value function bonds a medium and a value, enables the determination of the value of electronic cash and the authenticity of monetary value upon evaluation. The holder function is defined as follows.

Definition 3. *For a certain medium m and an owner of electronic cash u, a holder function is a mapping function of a medium to a holder. This notion is declared as:*

$$hf_m^u = f(m, u).$$

The holder function represents proof of ownership of an electronic cash data. Any entity can define the ownership of an electronic cash by using the holder function. By using Definition 2 and Definition 3, we can redefine the definition of electronic cash data (from Definition 1) as

$$e = f(vf_m^v, hf_m^u). \tag{1}$$

The rest of the model uses Equation (1) as reference to electronic cash data. Equation (1) implies that an electronic cash data must, at least, consist of two data: Value function and holder function.

2.2 Electronic Cash System

The sub-model of electronic cash system describes the entities, the processes of electronic cash, and the relationship between entities and processes. First, we need to define the entities in electronic cash system. In most of the schemes such as in [4] and [6], the schemes involve three entities. However, there are schemes which involve more entities (such as [23]) or less [3].

In this model, we model the entities as a set. There are four entities in the set. The role in electronic cash system defines each entity. We describe the model of entity as follows.

Definition 4. *Let E be a set of entity in an electronic cash scheme. The set of E is defined as*

$$E = \{P, U, I, R\},$$

where:

- *P is a single principal that manage and arbitrate the entire system,*
- *I is an issuer of electronic cash,*
- *U is a finite set of electronic cash user/holder, where $U = u_1, u_2, \ldots, u_i, \forall i \in \mathbb{Z}^+$,*
- *R is a finite set of electronic cash merchant/receiver, where $R = r_1, r_2, \ldots, r_j, \forall i \in \mathbb{Z}^+$.*

It is possible for a P and I to be implemented as a single entity. This can be seen in [19]. Although in [18] there seems to exist separate I and P, both original signer and proxy signer can be considered as the same entity with the responsibility of both I and P.

The processes of electronic cash defines how the system works. Each process involves one or more entities. We define the model of process as a set of processes as follows.

Definition 5. *Let A be a set of action/process related to electronic cash system. The membership of A is defined as:*

$$A = \{\texttt{SETUP}, \texttt{CREATE}, \texttt{SPEND}, \texttt{DEPOSIT}, \texttt{ARBITRATE}\},$$

where:

- `SETUP` *is a process to generate system parameters, including entity's credential,*

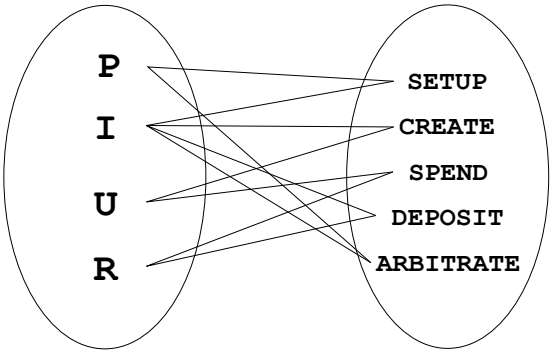


Figure 1: The mapping of electronic cash system

- `CREATE` *is a process to generate electronic cash data* e,
- `SPEND` *is a process to use an electronic data* e *in a transaction,*
- `DEPOSIT` *is a process to settle electronic cash data usage,*
- `ARBITRATE` *is a process to settle a dispute regarding electronic cash usage.*

The processes in Definition 5 represent the general process of an electronic cash system. It only describes the processes directly linked to electronic cash system's life cycle [5]. A scheme may implement more process, such as in [3]. However, these extra process are usually a sub process of a process in A.

Interaction between entities and process related to electronic cash is a general definition of electronic cash system. By referring to Definition 4 and Definition 5, we can redefine the general definition as "*a set of entities conduction a set of processes to use electronic cash data*". This definition can uses a simple mapping as its model, as shown in Figure 1. The formal definition of electronic cash system can be defined as follows.

Definition 6. *An electronic cash system* $\hat{e}$ *is a many-to-many mapping from set* E *to set* A *over finite amount of electronic cash data*

$$\hat{e} : E \to A.$$

2.3 Electronic Cash Form

The Definition 1 and Equation (1) define the data of electronic cash. However, the implementation method of value function determines the form of e-cash. In scheme such as [16], each e has a single denomination value, that will not change throughout its life cycle. Different approach is taken by some scheme, such as [4]. The value of e can change during its life cycle.

The first case the value function is constant. The value function will not change on a transaction, even when the holder function changed. The form which implements this method is defined as *fix-valued electronic cash*. The definition of this form is defined as:

Definition 7. *Let* v_a *be a constant that represent a denomination, where* $v_a \in \mathbb{Z}^+$, *and* m_a *is a single unique medium which holds* v_a. *A fix-valued electronic cash is a system of* $\hat{e}$ *that satisfy:*

$$vf_{m_a}^{m_a} = f(m_a, v_a) = constant$$

for any n `SPEND` *operation.*

The second case of implementation changes the electronic cash value function but usually retain its holder function. This form of electronic cash can be defined as *variable-valued electronic cash*. The formal definition is as follows.

Definition 8. *For a given medium* m_a, *a variable-valued electronic cash is a system of* $\hat{e}$ *that satisfy:*

$$vf_{m_a}^{v} = vf_{m_a}^{v_1} + vf_{m_a}^{v_2} + \ldots + vf_{m_a}^{v_n}$$

or

$$vf_{m_a}^{v} = f(m_a, v(n))$$

where n *is any number of* `SPEND` *operation, and* $v_1, v_2, \ldots, v_n \in \mathbb{Z}^+$.

3 Model of Electronic Cash Processes

The sub model of electronic cash processes defines the models of each process in Definition 5. The model contains the algorithm of each process. This algorithm explain how to conduct each process in general terms. The implementation of this model may contains more steps and protocols, depends on the underlying cryptographic scheme or mechanics in the implementation.

3.1 SETUP Process

`SETUP` is a process to create a set of parameters as a basis of the entire system operation. The process involves P, I, or $u \in U$ to cooperate and create the parameters. The parameters could be in form of modulus, public-private key pair, or other value. The model of this process is as follows.

Definition 9. *For each entity* i, *where* $i \in E$, `SETUP` *is an algorithm to create a set of variables* $a_i = \{a_{i1}, a_{i2}, \ldots, a_{in}\}$ *as* i*'s parameters in system* $\hat{e}$. *Algorithm 1 explains this process.*

3.2 CREATE Process

`CREATE` defines the process of electronic cash withdrawal. In many schemes, the process is called withdraw process. This paper uses 'create' as this process name to emphasis the process of data creation.

Algorithm 1 SETUP Algorithm

1: Begin
2: i contacts P and request for admission in $\hat{e}$.
3: P calculates a_i.
4: P sends a_i to i.
5: i keeps a_i.
6: End

Definition 10. *For each $u \in U$,* CREATE *is a process to request an electronic cash data e with value of v from I by using steps explained in Algorithm 2.*

Algorithm 2 CREATE Algorithm

1: Begin
2: u choose a medium m_u.
3: u create a request in form of $r_e = f(a_u, m_u, v)$ and send it to I.
4: I calculate $vf^v_{m_u}$, $hf^u_{m_u}$, and $e = f(vf^v_{m_u}, hf^u_{m_u})$.
5: I send e to u and deduct an amount of v from u's account.
6: End

In some scheme, such as [11, 13], CREATE process is not an independent process. It exist as a part of SETUP process. In this scheme, a medium m is a part of user u's system parameter a_u. The electronic cash data e can be used multiple times, with each usage has a value of fixed v. With these conditions, this scheme still fulfill Definition 1 and Definition 10.

3.3 SPEND Process

This part of the model describes the core process of electronic cash system, which is the exchange of e. This process only involves a user u and a merchant r. The definition of this model is as follows.

Definition 11. *For a pair of user $u \in U$ and a receiver $r \in R$,* SPEND(u, r, e) *is an operation to exchange electronic cash data e from u to r by using Algorithm 3.*

Algorithm 3 SPEND Algorithm

1: Begin
2: u and r agree on a value v.
3: u send $e = f(vf^v_m, hf^u_m)$ to r.
4: **if** r verify that $f(m, v) = vf^v_m$ AND $f(m, u) = hf^u_m$ AND $vf^v_m \geq v$ **then**
5: r create a receipt $r_t = f(u, r, e, a_r)$.
6: **else**
7: r reject and abort process.
8: **end if**
9: End

Some scheme may delegate the verification process in Algorithm 3 to I. In this scheme, r simply contact I and send the transaction data (e, u, r, a_r) to I (this step usually found at on-line scheme). Since in the end r still receive the result of verification and decide to continue or not, the process still fulfill Definition 11.

3.4 DEPOSIT Process

An electronic cash data life cycle when it is returned to I. DEPOSIT process explains the steps to terminate an electronic cash data usage. This process can be considered as a process to change electronic cash to cash, or an opposite process of CREATE. The definition of this process is as follows.

Definition 12. *For each e received by a $r \in R$,* DEPOSIT *is an operation between r and I to settle the usage of e by using Algorithm 4.*

Algorithm 4 DEPOSIT Algorithm

1: Begin
2: r send I a set of electronic cash data $e_1, e_2, \ldots, e_n$, where n is the number of electronic cash data to be settled.
3: **for all** e in set **do**
4: **if** I verify $vf^v_m = f(v)$ AND $vf^v_m \notin L_e$, where L_e is a list of used e **then**
5: I add v to r account.
6: I add e_n to list L_e so that $L_e \cup e_n$.
7: **else**
8: I reject e_n.
9: **end if**
10: **end for**
11: End

I may find a data that has been used before or formed without proper protocol. After finding such data, I can use ARBITRATE process to track the responsible user.

3.5 ARBITRATE Process

Some dispute may arise from the usage of electronic cash. A user may forges a data and uses it on a transaction. A merchant may receives a double spent electronic cash. A dishonest issuer may accuses honest user of doing double spend. To settle these disputes, we need to prove two things. First we need to validate the electronic cash data, by proving the value function and holder function. Second is to determine the adversary identity.

ARBITRATE models the process to determine the validity of e or to identify of an adversary. The model is defined as follows.

Definition 13. *For a dispute between any entity $i \in E \rightarrow i \neq P$ over a transaction of e,* ARBITRATE *is an operation conducted by P to determine the usage of e or to trace any entity involved with e by using Algorithm 5.*

Algorithm 5 ARBITRATE Process

Begin
An entity i request an arbitration to P.
i send data of the disputed transaction (e, r_t).
if P verify $f(m, v) = vf_m^v$ AND $vf_m^v \notin L_e$ **then**
  e is valid.
else
  e is forged or double spent.
end if
if P verify $r_t = f(u, r, e)$ **then**
  The transaction between u and r is valid.
else
  The transaction is not valid.
end if
if The user $u \in f(m, u) = hf_m^u$ **then**
  P prove the ownership of u over e.
else
  u is not the owner of e.
end if
End

4 Model of Electronic Cash Properties

We present a list of property model commonly found in electronic cash scheme. The property can be divided into two general categories: functional, and security. It is not mandatory to implement all properties in a scheme. However, two security properties must exist for a scheme to be functional.

4.1 Functional Property

Functional property model covers all property that can help the operation of electronic cash system. This type of property is not mandatory, a scheme can operates appropriately without a functional property. However, some scheme may gains additional benefit by implementing this property. The model of electronic cash functional property consist of *divisibility, peer-to-peer*, and *transferable*.

Divisibility describes the behavior of electronic cash data value function. A divisible electronic cash data can be used multiple time by an user without changing its medium. The value function of electronic cash with this property can be divided into smaller value. We define divisibility as follow:

Definition 14. *Divisible electronic cash is a system of $\hat{e}$ where for each e, the value function for a certain medium vf_m^v is a sum of arbitrary smaller values v_n. Each v_n can be used in any n transaction by the same $u \in U$. The value function of divisible electronic cash must satisfy:*

$$vf_m^v = vf_m^{v_1} + vf_m^{v_2} + \ldots + vf_m^{v_n},$$

where

$$v = v_1 + v_2 + \ldots + v_n.$$

Definition 14 also complies to Definition 8. This means that divisible electronic cash has the form of variable-valued electronic cash. It is also means that variable-valued type of e-cash always has the divisible property.

Peer-to-peer is a property that describes the implementation behavior of SPEND process. As we have stated before, the verification of e can be delegated to I. If a scheme can use SPEND without involving any entity beside r and u, the scheme has the property of peer-to-peer. The complete definition is as follows.

Definition 15. *A system $\hat{e}$ is a peer-to-peer electronic cash system if for any* SPEND *process there is an ordered pair with an exact member, such as* $\{(U, \texttt{SPEND}), (R, \texttt{SPEND})\}$.

By Definition 11, an user u transfers electronic cash to r without changing its holder function. This electronic cash data cannot be used in another transaction by r and it must be settled by using DEPOSIT process. Transferable property alter this behavior, it enables the transfer of electronic cash data ownership by alters its holder function. The receiver can used the electronic cash data in another transaction. The definition of this property is as follows.

Definition 16. *A system $\hat{e}$ is having a transferable property if for all e there can there is a* SPEND *process between two user, u_1, u_2 and $u_1 \neq u_2$, so that the process fulfill:*

$$f(vf_{m_t}^{v_t}, hf_{m_t}^{u_2}) = f(vf_{m_2}^{v_2}, hf_{m_2}^{v_2}) + f(vf_{m_1}^{v_1}, hf_{m_1}^{u_1}),$$

where t denotes the time after SPEND *process,*

$$\begin{aligned} v_t &= v_2 + v_1, \\ m_t &= m_2 + m_1. \end{aligned}$$

Transferable property simplify the entity set E. In a system with transferable property, it is applies that $U = R$. There is no need to set up the parameter of different group of entity thus reducing the system complexity. The example of scheme with this property can be found in [1].

4.2 Mandatory Security Property

Within any monetary system, forgery poses significant threat to the entire system. In electronic cash system, a user who able to forge electronic cash data can generate any number of data without the proper process. As a result, the system cannot be trusted for further operation. Therefore, the property of unforgeability is a must. We define unforgeabiltity as follows.

Definition 17. *A system $\hat{e}$ is said to have unforgeability if $\forall u, r \in E$ there is no non-negligible advantage to form a valid e without using* CREATE *process.*

Double spending is an action where a user, or a receiver, uses a value function more than once in a different transaction (SPEND process). This action is a variation of forgery. However, if in forgery the value function and

holder function is not valid data, in double spend both value is a valid data made by a proper process.

A system of $\hat{e}$ must have a mechanism to detect double spent data to prevent (or to search for the perpetrator) double spending. For example, using the list L_e from Algorithm 4, an entity can determine whether a data has been used before or not. To model this property in more formal manner, we define the property of double spending prevention as follows.

Definition 18. *A system $\hat{e}$ is said to have double spending prevention property if $\forall u \in U$ there is no non-negligible advantage to execute two or more* `SPEND` *process to any $r_1, r_2 \in R$ with the same $e = f(vf_m^v, hf_m^u)$. Or, $\forall r \in R$ there is no non-negligible advantage to execute two or more* `DEPOSIT` *process for a single $e = f(vf_m^v, hf_m^u), \forall u \in U$.*

The advantage described in Definition 17 and Definition 18 not only refers to the probability of success of forgery or double spending. The phrase also refers to the feasibility of the actions. If a scheme has non-negligible probability but infeasible to do the forgery or double spending, the adversary is considered to have negligible advantage to do the action.

4.3 Optional Security Property

The optional security properties are not mandatory, such as unforgeability and double spending prevention. The electronic cash system still secure in the absence of these properties. However, the implementation of these properties will add additional layer of security into the scheme.

The first property is *anonymity*. The works of Cannard-Gouget classify anonymity into 4 different levels: weak, strong, full, and perfect anonymity [2]. However, the notion of anonymity of Cannard-Gouget classification merges anonymity with *unlinkability*. To prevent the confusion between the two notions, we models anonymity with unlinkability separately. The fulfillment of Cannard-Gouget classification in this model, depends on the fulfillment of anonymity and unlinkabiltiy in this paper. In this paper, we define anonymity as follows.

Definition 19. *Let $r_t(e)$ be a transaction receipt of a certain* `SPEND` *process between $u \in U$ and $r \in R$. A system $\hat{e}$ has the property of anonymity if for any entity $i \in E, i \neq \{u, r, P\}$, there is no non-negligible advantage to determine that $\{u, r\} \in r_t(e)$.*

Unlinkability is a property that ensure that no one can track the movement of electronic cash data. If an adversary can see two distinct transactions, he/she shall not be able to link the two transaction to a user (even if both transaction involve the same user). We define the unlinkability as follow:

Definition 20. *Let e_1, e_2 be two distinct electronic cash data owned by $u \in U$, and r_1, r_2 be the transaction receipt of e_1, e_2 respectively. A system $\hat{e}$ has the unlinkability property if for any entity $i \in E, i \neq \{u, P\}$, there is no non-negligible advantage to determine that:*

$$\{u\} \in r_1(e_1) \cap r_2(e_2),$$

where

$$\begin{aligned} e_1 &= f(hf_{m_1}^u), \\ e_2 &= f(hf_{m_2}^u). \end{aligned}$$

The last security property related to the common assumption in electronic cash scheme. Many scheme assume that entity I is trusted by the entire system. It is assumed that I will not deliberately conduct any action that disadvantageous to another honest entity.

The *exculpability* property disregards this assumption of I. A scheme that has exculpabilty property (such as [25]) deploys a mechanism to ensure that I can be trusted. The exculpability property prevent I to accuse an honest user of double spending. It also prevent I to create e without any request from u. We define exculpability as follows.

Definition 21. *A system $\hat{e}$ have exculpability property if for any honest user $u \in U$ and a non-exist electronic cash data e_x, there is no non-negligible advantage for I to claim that $e_x \in L_e \leftrightarrow$* `SPEND`*$(u, r, e_x)$ or that $e_x = f(hf_m^u)$.*

5 Using The Model for Security Analysis

At this section, we will use our model to analyze the security of an existing electronic cash scheme. We use this activity to validate our model. We aim to analyze the security of Chaum's Untraceable Electronic Cash scheme [6] using our model.

As a starting point, we define the electronic cash data e in this scheme (from this point, we will refer Chaum's scheme as "scheme"), which in form of

$$C = \prod_{1 \leq i \leq k/2} f(x_i, y_i)^{1/3} mod\ n. \tag{2}$$

The identity of a user is represented by an account number u and a counter v. Both values are components of $y_i = g(a_i \oplus (u||(v+i)))$. Each C has a fixed denomination of v. These conditions fulfill Definition 1 and Equation (1) to describe e. It can also be noted that the scheme fulfill Definition 7, which make it a fix-valued electronic cash.

The electronic cash data in the scheme is made by using withdraw protocol between a user and the bank (issuer of electronic cash data, hence I). The steps of this process can be summarized as follows.

From Algorithm 6, we can see that all the variables needed to construct C is made by the user. The bank only verify the ownership of B_i and debit the user account. There is no process that involve the bank secret parameter in the construction of C. With this condition,

Algorithm 6 Withdraw Process of [6]

1: Begin
2: The user choose a_i, c_i, d_i and r_i, where $1 \leq i \leq k$, randomly from residue of *mod* n.
3: The user send $B_i = r_i^3.f(x_i, y_i) mod\ n$, for all i, where $x_i = g(a_i, c_i)$ and $y_i = g(a_i \oplus (u||(v+i)))$ to bank.
4: The bank choose $R = \{i_j\}$, where $1 \leq i_j \leq k, 1 \leq j \leq k/2$.
5: **for all** $i \in R$ **do**
6: The user send a_i, r_i, c_i, d_i to the bank.
7: **end for**
8: The bank send the user $\prod_{i \notin R} B_i^{1/3} = \prod_{1 \leq i \leq k/2} B_i^{1/3} mod\ n$.
9: The user extract the electronic cash data $C = \prod_{1 \leq i \leq k/2} f(x_i, y_i)^{1/3} mod\ n$.
10: End

the user actually can produce C without using the withdraw protocol with great probability. The user only needs to choose a set of a_i, c_i, d_i, and r_i and construct C using step 9 in Algorithm 6. Therefore, the scheme is not fulfilling Definition 17.

The scheme need to fulfill the second mandatory property of security, the double spending prevention. The double spending prevention mechanism could be analyzed from the scheme's `SPEND` and `DEPOSIT` action. In the scheme, both action are combined into one protocol. The protocol is summarized in Algorithm 7.

Algorithm 7 Spend Protocol of [6]

1: Begin
2: The user u send C to the receiver r.
3: r choose a binary string, $z_1, z_2, \ldots, z_{k/2}$ and send it to u.
4: **for all** z_i in binary string **do**
5: **if** $z_i = 0$ **then**
6: u send $x_i, a_1 \oplus (u||(v+i))$ to r.
7: **else**
8: u send a_i, c_i, y_i to r.
9: **end if**
10: **end for**
11: **if** r can verify the correctness of C **then**
12: r accept C.
13: **else**
14: r reject C.
15: **end if**
16: r send C, the binary string $(z_1, z_2, \ldots, z_{k/2})$, and all u's responses to I.
17: **if** I can verify the transaction **then**
18: I credit r account.
19: **else**
20: I reject the transaction.
21: **end if**
22: End

Step 16 to 21 in Algorithm 7 can be executed separately from the rest of steps. These steps represent the `DEPOSIT` action in the scheme. The receiver may wait until end of the day to execute theses steps for all transaction he/she receives in the day. This delay may result in a double spending attempt by the user.

According to Algorithm 7, the receiver cannot check whether a data has been used before by the same user. Therefore, it is quite possible for a user to spend a data in a receiver, then uses the same data in another transaction with another receiver. However, using step 16 to 21, I can detect the double spender quite easily and run a tracing algorithm to determine the user identity. If a user double spend an electronic cash data, then I will, with great probability, acquires both a_i and $a_i \oplus (u||(v+i))$ components of the same i in the electronic cash data. By using XOR operation on both components, I can extract $(u||(v+i))$ which contains the user's identity u. In short, it is quite improbable for an u to conduct a double spend without being detected and traced by I. Thus, the scheme fulfill Definition 18.

From Algorithm 7, we can also see that the `SPEND` process involves two entities: the user and the receiver. The receiver validates the electronic cash data without the help of another entity. This condition fulfill the description in Definition 15, which make the scheme has the property of *peer-to-peer*.

6 Comparison with Another Model

As we have stated earlier, we build our model based on the model of Inenaga *et al.* [15]. We find that the model in Inenaga *et al.* only covers a portion of electronic cash system. The model can only be used to describe an off-line electronic cash (peer-to-peer) involving only two entities. The complete comparison between our model and Inenaga *et al.* model can be seen in Table 1.

The model proposed by Inenaga *et al.* cannot be used to model scheme such as find in Kang & Xu [16], even when the scheme uses off-line transaction. The Kang & Xu scheme involve entity such as Bank and Trustee, which is not described in Inenaga *et al.* model. The scheme of Kang & Xu is built with property of anonymity as its goals, which is not found in the model of Inenaga et al. At best, the model of Inenaga *et al.* can only models small part of Kang & Xu's scheme.

On the contrary, Bank and Trustee is covered in our model as Issuer and Principal. Our model also provide a property model that can explain anonymity. Compared to Inenaga *et al.* model, our model can easily models the entire scheme of Kang & Xu. This illustrate our model's capability to model a wider range of electronic cash scheme compared to Inenaga *et al.* model.

Table 1: Comparison of models

Model	SubModel of System	SubModel of Process	SubModel of Property
Inenaga *et al.*	Money System E-money Type	Money Transfer	Money Forgery Forged Money Transfer Detectability of Forged Money
Proposed	Electronic cash data Electronic cash system Electronic cash form	SETUP CREATE SPEND DEPOSIT ARBITRATE	Functional Property Security Property

7 Conclusions

The proposed model has more comprehensive approach compared to the model in [15]. Figure 2 shows the resume of our model. We divide the model into 3 sub models: the model of system, the model process, and the model of properties. Due to its generality, our model can be used to describe most of existing electronic cash scheme. However, it cannot be used to describe a specific mechanics used in specific scheme. For example, the model of `SPEND` process cannot describe the process of updating electronic cash record to the entire system in scheme [25].

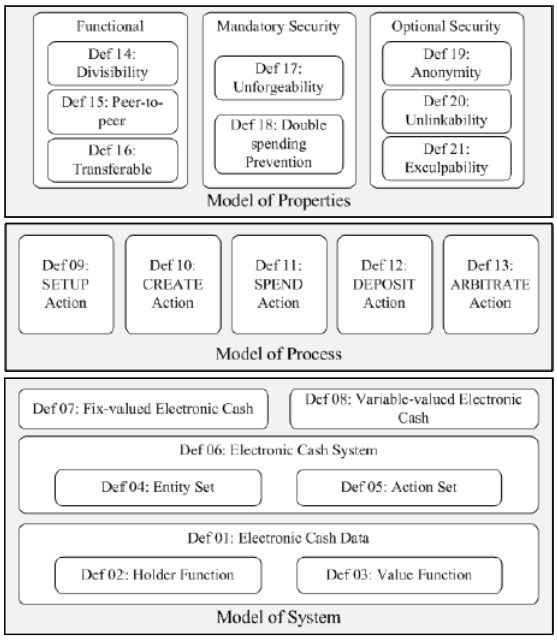


Figure 2: The proposed model

The proposed model can be used as a helping tool to build new electronic cash scheme. The model can be considered as a skeleton to build more detailed scheme. Any person can use the model as a reference on how electronic cash should behave. By using the security property model, the builder of the scheme can ensure that their scheme has the proper security mechanism.

As we have shown in previous example, our model is suited as a reference to compare or evaluate existing scheme. By using our model, any method of evaluation, such as in [7, 9, 28], can have a clear definition on determining the performance of the evaluated scheme. By having a clear definition of security objective, we can avoid mistakes because of the difference in security definition.

We also believe that our model can be used to help the development of payment scheme on a specific platform but not necessarily electronic cash scheme, such as scheme in [10, 12, 17, 24]. Although, the two scheme does not explicitely uses electronic cash, it has the same fundamental principle. The electronic cash and these payment schemes have more similarity compared to electronic cash and cryptocurrency.

This model does not cover distributed electronic cash or cryptocurrency, such as Bitcoin, due to the difference in underlying mechanism. However, we find that the basic principle of centralized and distributed electronic cash is the same. For example, both centralized and distributed electronic cash must have unforgeability and double spending prevention property. It is interesting to expand this model to cover distributed electronic cash

There are many attempt to use centralized electronic cash scheme as a mixing agent to increase the anonymity property of distributed electronic cash scheme. Scheme such as [8, 14, 20, 21, 27], using centralized electronic cash mechanism to create a masking medium to Bitcoin. These schemes has more similarity to centralized electronic scheme while operating under the paradigm of distributed electronic cash. These schemes is suitable as the first stepping stone to develop more general model that covers distributed electronic cash.

References

[1] O. Blazy, S. Canard, G. Fuchsbauer, A. Gouget, H. Sibert, and J. Traoré, "Achieving optimal anonymity in transferable e-cash with a judge," in *International Conference on Cryptology in Africa*, pp. 206–223, 2011.

[2] S. Canard and A. Gouget, "Anonymity in transferable e-cash," in *International Conference on Applied Cryptography and Network Security*, pp. 207–223, 2008.

[3] S. Canard, A. Gouget, and J. Traoré, "Improvement of efficiency in (unconditional) anonymous transferable e-cash," in *International Conference on Financial Cryptography and Data Security*, pp. 202–214, 2008.

[4] S. Canard, D. Pointcheval, O. Sanders, and J. Traoré, "Divisible e-cash made practical," *IET Information Security*, vol. 10, no. 6, pp. 332–347, 2016.

[5] D. Chaum, "Blind signatures for untraceable payments," in *Advances in cryptology*, pp. 199–203, 1983.

[6] D. Chaum, A. Fiat, and M. Naor, "Untraceable electronic cash," in *Conference on the Theory and Application of Cryptography*, pp. 319–327, 1988.

[7] Y. Chen and J. S. Chou, "On the privacy of user efficient recoverable off-line e-cash scheme with fast anonymity revoking," *International Journal Network Security*, vol. 17, no. 6, pp. 708–711, 2015.

[8] G. Danezis, C. Fournet, M. Kohlweiss, and B. Parno, "Pinocchio coin: Building zerocoin from a succinct pairing-based proof system," in *Proceedings of the First ACM workshop on Language Support for Privacy-enhancing Technologies*, pp. 27–30, 2013.

[9] J. Dreier, A. Kassem, and P. Lafourcade, "Formal analysis of e-cash protocols," in *12th International Joint Conference on e-Business and Telecommunications (ICETE'15)*, vol. 4, pp. 65–75, 2015.

[10] T. H. Feng, M. S. Hwang, and L. W. Syu, "An authentication protocol for lightweight NFC mobile sensors payment," *Informatica*, vol. 27, no. 4, pp. 723–732, 2016.

[11] M. S. Hwang, C. C. Lee, Y. C. Lai, "Traceability on low-computation partially blind signatures for electronic cash", *IEICE Fundamentals on Electronics, Communications and Computer Sciences*, vol. E85-A, no. 5, pp. 1181–1182, May 2002.

[12] M. S. Hwang, I. C. Lin, L. H. Li, "A simple micropayment scheme", *Journal of Systems and Software*, vol. 55, no. 3, pp. 221–229, Jan. 2001.

[13] M. S. Hwang and P. C. Sung, "A study of micropayment based on one-way hash chain," *International Journal Network Security*, vol. 2, no. 2, pp. 81–90, 2006.

[14] M. H. Ibrahim, "Securecoin: A robust secure and efficient protocol for anonymous bitcoin ecosystem," *International Journal Network Security*, vol. 19, no. 2, pp. 295–312, 2017.

[15] S. Inenaga, K. Oyama, and H. Yasuura, "Towards modeling stored-value electronic money systems," *Information and Media Technologies*, vol. 6, no. 1, pp. 25–34, 2011.

[16] B. Kang and D. Xu, "Secure electronic cash scheme with anonymity revocation," *Mobile Information Systems*, vol. 2016, 2016.

[17] I. C. Lin, M. S. Hwang, C. C. Chang, "The general pay-word: A micro-payment scheme based on n-dimension one-way hash chain", *Designs, Codes and Cryptography*, vol. 36, no. 1, pp. 53–67, July 2005.

[18] J. Liu and Y. Hu, "A new off-line electronic cash scheme for bank delegation," in *5th International Conference on Information Science and Technology (ICIST'15)*, pp. 186–191, 2015.

[19] J. W. Lo, H. M. Lu, T. H. Sun, and M. S.Hwang, "Improved on date attachable electronic cash," *Applied Mechanics and Materials*, vol. 284, pp. 3444–3448, 2013.

[20] I. Miers, C. Garman, M. Green, and A.D. Rubin, "Zerocoin: Anonymous distributed e-cash from bitcoin," in *IEEE Symposium on Security and Privacy (SP'13)*, pp. 397–411, 2013.

[21] K. Naganuma, M. Yoshino, H. Sato, and T. Suzuki, "Auditable zerocoin," in *IEEE European Symposium on Security and Privacy Workshops (EuroS&PW'17)*, pp. 59–63, 2017.

[22] S. Nakamoto, "Bitcoin: A peer-to-peer electronic cash system," *Bitcoin*, 2008. (`https://bitcoin.org/en/bitcoin-paper`)

[23] H. Oros and C. Popescu, "A secure and efficient off-line electronic payment system forwireless networks," *International Journal of Computers Communications & Control*, vol. 5, no. 4, pp. 551–557, 2010.

[24] H. H. Ou, M. H. Hwang, and J. K. Jan, "A provable billing protocol on the current umts," *Wireless Personal Communications*, vol. 55, no. 4, pp. 551–566, 2010.

[25] T. Sander and A. Ta-Shma, "Auditable, anonymous electronic cash," in *Annual International Cryptology Conference*, pp. 555–572, 1999.

[26] D. E. Saputra and S. H. Supangkat, "A study of electronic cash paradigm," in *International Conference on Information Technology Systems and Innovation (ICITSI'14)*, pp. 273–278, 2014.

[27] E. B. Sasson, A. Chiesa, C. Garman, M. Green, I. Miers, E. Tromer, and M. Virza, "Zerocash: Decentralized anonymous payments from bitcoin," in *IEEE Symposium on Security and Privacy (SP'14)*, pp. 459–474, 2014.

[28] F. Wang, C. C. Chang, and C. Lin, "Security analysis on secure untraceable off-line electronic cash system.," *International Journal Network Security*, vol. 18, no. 3, pp. 454–458, 2016.

Biography

Dany Eka Saputra received B.Eng. in Aeronautics and Astronautics from Institut Teknologi Bandung in 2007. He also obtained his M.Eng. in Electrical Engineering from the same university in 2012. Currently, he is taking his Doctoral Study in Electrical Engineering and Informatics from Institut Teknologi Bandung. He also a faculty member of Departement of Informatics at STMIK "AMIKBANDUNG". His main interest is information security with specialization in electrical cash security. His other interests include game tecnology and protocol

engineering.

Sarwono Sutikno received B.Eng. in Electronics from Institut Teknologi Bandung in 1984. He then received his Dr.Eng. in Integrated System from Tokyo Institute of Technology in 1994. Currently, he is an Associate Professor at Institut Teknologi Bandung. An active member of ISACA with several certification. His main interests are information security and cyber security.

Suhono harso Supangkat received his B.Eng. in Electrical Engineering from Institut Teknologi Bandung in 1986. He received his Dr.Eng. in Information System Science from Tokyo University of Electro-communication in 1998. Currently, he is a Professor at Institut Teknologi Bandung. His main interest is smart city system and technology. Actively promoting smart city concept and technology from 2012.

Secure Traffic Efficiency Control Protocol for Downtown Vehicular Networks

Maram Bani Younes
(Corresponding author: Maram Bani Younes)
Computer Science Department, Philadelphia University, Jordan
P. O. Box: 19392 V Amman - Jordan
(Email: mbani047@uottawa.ca)
(Received Nov. 6, 2017; Revised and Accepted June 18, 2018; First Online Feb. 26, 2019)

Abstract

The intelligent transport system increasingly considers the traffic efficiency applications over the road networks. This type of application aims mainly at reducing the traveling time of each vehicle toward its targeted destination/destinations and deceasing the fuel consumption and the gas emissions there. The Vehicular Ad-Hoc Networks (VANETs) technology is one of the main approaches that have been used in these applications. However, the connecting environment of VANETs introduces a good chance for malicious drivers to take advantages of other cooperative drivers and deceive them to achieve their own benefits. This paper introduces a Secure Traffic Efficiency control Protocol (STEP). The designed protocol means to secure the traffic efficiency control applications over the downtown areas. It protects the privacy of cooperative drivers and minimizes any damage that malicious drivers may cause. From the experimental results, the STEP protocol succeeds to detect malicious nodes over the road network. Thus, it enhances the correctness of the traffic efficiency applications and increases their feasibility.

Keywords: Authentication; Integrity; Malicious; STEP; Traffic Efficiency

1 Introduction

Several protocols have been introduced recently aiming to efficiently use the available resources over the downtown road scenarios [4,12,20,32]. The grid-layout of the modern downtown areas directs the researchers to develop protocols that first evaluate the real-time traffic characteristics of each road segment separately [34,39]. Then, according to the traffic distribution over the downtown area, several research studies have selected the best path toward any targeted destination in terms of traveling time [38], fuel consumption and gas emission [25, 40] or the context of each road segment [33]. Moreover, located traffic lights are significant in term of controlling the traffic efficiency. Several protocols have been introduced aiming to intelligently schedule the phases of each traffic light based on the traffic distribution over the neighboring road segments [23, 36, 37].

Several security issues are threaten the traffic efficiency protocols over the downtown areas [18]. Indeed, these issues have dangerous consequences when attackers exploit the venerabilities in the traffic efficiency protocols. Malicious attackers can be categorized into four main groups according to their targets: Vandal, selfish, intruder, and prankster. Vandal attackers aim to overload the network with useless packets, which causes losing important data and decreasing the functionality of the connecting network. Selfish attackers deceive other drivers, in order to achieve their own benefits while falsely direct the traffic. Intruders try to chase and stalk other drivers and their end destinations. finally, pranksters and criminals may try to deceive drivers in a certain area aiming to kidnap or hurt them.

In this work, we introduce a secure traffic efficiency control protocol (STEP) for downtowns, using the communication technology of VANETs. This protocol aims mainly to achieve the authenticity and the integrity of the transmitted data. Thus, it guarantees the correctness of the targeted efficiency control factor (*i.e.*, traveling time, traveling speed, fuel consumption, gas emission, *etc.*)

The remaining of this paper is organized as follows: In Section 2, we investigate some traffic efficiency control protocols and other traditional secure protocols that have been introduced using the communication technology of VANETs. We then define the general adversary threats of traffic efficiency protocols in Section 3. Next, the details of the secure traffic efficiency control protocol (STEP) is presented, in Section 4. After that, we present the experimental study which evaluates the efficiency, accuracy, and correctness of the STEP protocol compared to other unsecure traffic control protocols in Section 5. Eventually, Section 6 presents the entire conclusion of this work.

2 Related Work

In this section, we investigate the details of some traffic efficiency control protocols that have been developed using VANETs for downtown areas. After that, we explore some traditional secure protocols that have been designed for VANETs.

2.1 Traffic Efficiency Control Protocols

Several protocols have been introduced in the literature to control the traffic efficiency over the road network [11,13, 20,23,28,33,34,36,38]. These protocols aimed to enhance traffic fluency of vehicles on the road network. This is by decreasing the traveling time, the fuel consumption and the gas emission. It is also by increasing the traveling speed of each vehicle toward its destination.

The grid-layout of the downtown areas motivates the researchers in this field to investigate and locate the highly congested road segments. Then, recommend drivers to avoid these congested road segments during their trips. On the other hand, intelligent scheduling algorithms have been introduced for the installed traffic lights on downtown areas. These algorithms aim to decrease the waiting delay time of each vehicle at the signalized road intersections. Some of these algorithms use the traffic distribution on the neighboring road segments. Others consider the estimated arrival time of competing traffic flows.

2.1.1 Traffic Congestion Detection

The existed traffic evaluation and congestion detection protocols are classified into two main categories: Sensor-based protocols [7, 21, 27] and vehicular-based protocols [13,28,34,39].

The sensor based protocols provide real-time and accurate congestion level estimation for each investigated road segment. However, in these protocols special and expensive equipments (*e.g.*, camcorders, inductive loop detectors, antennas, radars, *etc.*) are required all over the area of interest. It is difficult to install and maintain these equipments regularly. Moreover, these equipments provide fixed-point or short-section traffic information limitations [21]. The basic traffic data is extracted from vehicles passing through the detection zone and saved for farther usage or analysis.

On the other hand, different traffic evaluation protocols have been introduced using the technology of VANETs. These protocols collect the basic traffic data of surrounding vehicles in each traveling zone. Traveling vehicles are expected to be equipped by VANETs-wireless transceiver and Global Positioning System (GPS) devices. Traveling vehicles broadcast their basic data periodically in order to announce their location, direction and speed during that period of time. Receiver vehicles can compute and/or predict the traffic density [13,39], traffic speed, or traveling time [42] of that area, using the gathered traffic data of the surrounding vehicles.

In order to expand the boundaries of the investigated area, Fukumoto *et al.* [13] used a blind forwarding mechanism where each vehicle forwards the received messages. On the other hand, Sankaranarayanan *et al.* [28] proposed a more efficient mechanism that forwards statistical data of the traffic situation over the area of interest. In our previously proposed work [34], we have introduced a protocol that specifically aimed to evaluate the traffic characteristics on any road segment in a downtown area. Based on the length of each road segment, reporting areas are virtually configured on that road where vehicles over these areas are responsible of forwarding the gathered traffic data. This mechanism aims mainly to deliver the traffic information between vehicles that cannot contact directly.

2.1.2 Road Traffic Control and Efficient Path Recommendations

Different protocols have been introduced to select the best path (*i.e.*, most efficient) toward each targeted destination. The grid-layout of the downtown area contains different paths that lead toward any targeted destination. Several protocols [4, 12, 20, 42] have used a central processor that gathers the real-time traffic distribution all over the investigated road network. The best or fastest path toward each targeted destination is obtained by the central processor. The best path recommendations are sent back to each traveling vehicle all over the area of interest. However, this centralized behavior introduces a bottleneck as well as single point of failure problems [30].

On the other hand, several researchers have designed a complete distributed path recommendation and congestion avoidance protocols [32, 33]. The best path toward each targeted destination is obtained and updated in a hop-by-hop fashion. The path is then constructed from the location of the targeted destination toward each road intersection all over the area of interest. Periodic and dynamic communications take its place among installed RSUs at each road intersection, in order to ensure full awareness of real-time traffic characteristics.

Furthermore, several protocols have been proposed to recommend the best path in terms of fuel consumption or gas emission [25, 40]. Other studies have considered the context of the road network [33] in terms of located services at each road segment. They aim to guarantee a certain level of congestion-free to special road segments (*e.g.*, a congestion-free level is guaranteed for road segments that lead to hospitals in order to allow the emergency cases to arrive it fast).

2.1.3 Intelligent Traffic Light Control

In order to design an intelligent scheduling algorithm for located traffic lights on downtown road networks, several mechanisms have been proposed. Some studies have introduced a scheduling algorithm for isolated traffic light (*i.e.*, single assumed traffic light) [15,16,24]. These studies have considered the real-time traffic characteristics of

competing flows of traffic at a single road intersection. Traffic volume and the length of vehicles' queues [31], traffic speed and density [36] and estimated arrival times [23] are the main real-time parameters that have been considered to obtain efficient schedules for isolated traffic lights.

Several other studies have considered the cooperative communications among located traffic lights over road networks [22, 37, 41]. These studies have produced scheduling algorithms for each traffic light located on close road network or open road networks. It is referred to the synchronized situation among located traffic lights on grid-layout road network where all road segments have the same priority to cross the signalized intersection as close road network [14]. On the other hand, the scenarios where an arterial street (*i.e.*, set of continues road segments) is existed on the road network vehicles over this street have a higher priority to cross any signalized intersection before conflicted traffic flows is referred to as open network [29].

For the open and close road network scenarios. Besides, considering the traffic characteristics of the competing traffic flows, the schedule of each traffic light in these scenarios have considered the estimated arrival platoons of vehicles from the neighboring road intersections [5,22,41]. The number of vehicles, traveling speed and estimated arrival time of each platoon are the main characteristics to consider in these algorithms.

2.2 Secure Protocols for VANETs

The high speed mobility and extended geographical area of the VANET technology have produced real challenges to secure the introduced applications there. Special mechanisms have been designed to enhance the secure communications on VANETs. Several studies have been introduced to guarantee the authenticity, integrity and confidentiality feature for VANET in general [9, 10, 17].

Recently, researchers start developing secure service protocols. These protocols provide a certain service and specifically considering the security requirements of that service. To mention a few, secure cooperative collision warnings [26], secure position information [3], secure information dissemination [1], and secure service discovery protocols [2]. In these studies, first an adversary model is defined specifically to the investigated application, then the security mechanism are developed to handle the defined venerabilities, to eliminate threats and to mitigate the risks there.

In our previous work [35] we have presented a secure traffic evaluation protocol (SCOOL). This protocol remarks the security threats of the traffic evaluation protocols on the downtown areas and introduces solutions for each defined vulnerability there. In this paper, we aim to expand our previous work to investigate the vulnerabilities of other traffic efficiency applications on the downtown areas such as: Path recommendations and traffic light controlling mechanisms. Then, a complete secure traffic efficiency control protocol for downtown areas is proposed, we name this protocol by STEP.

3 Adversary Model of The Traffic Efficiency Control Protocols

The traffic efficiency is one of the main categories in the vehicular network applications. Evaluating the real-time traffic characteristics of the road network. Recommending vehicles to follow the most efficient path toward their targeted destinations. Scheduling the located traffic lights according to the real-time traffic distribution on the competing traffic flows. Many other applications have been proposed aiming mainly to increase the traffic fluency and efficiently use the available resources over the road network. All of these applications vitally require the traffic reports of traveling vehicles. Cooperative communications among traveling vehicles and installed road-side units (RSUs) help to gather the real-time traffic characteristics of the investigated area of interest. These traffic characteristics are processed and analyzed to obtain the most efficient recommendations for drivers, traffic lights and other road components. In this section, we discuss three main adversaries on traffic efficiency control protocols.

1) **Integrity**: Aims to ensure that data has not been altered by unauthorized users. It also prevents accidental hold or deletion of data by users. Three main threats can be categorized under this adversary:

   a. *Forgery*: Some drivers alter the reported speed or location of their vehicles. Then, the vital message of evaluating the traffic characteristics of each area of interest carries wrong data. Moreover, vehicles that forward messages toward far areas may also alter and compromise the forwarded messages or initiate a fake report. This causes to generate inaccurate traffic evaluation reports for the road network. Then, it reduces the performance and correctness of the corresponding efficiency control protocol such as efficient path recommendation protocols and intelligent traffic light scheduling algorithms.

   b. *Denial of service*: In this case, attackers forbade the communication channel by overloading it with useless messages. Attackers can use the Botnet system (*i.e.*, set of compromised nodes attack the same target on the computer network). Unexpected large number of fake vehicles asking for recommendations from the same RSU prevent other vehicles from sending their requested information. Several vehicles broadcast large number of messages in a short period of time increases the demand on the communication channels as well. These scenarios negatively affect the performance and accuracy of traffic efficiency application protocols.

c. *Black-hole attack*: Some attackers and malicious vehicles drop all or few selected packets without informing the senders. Then, several packets are lost over the network and will not be considered in the traffic evaluation. Based on the importance and number of the lost packets, this affects the performance of the traffic efficiency control protocols.

2) **Impersonation**: Is used to gain an access to the vehicular network in order to commit fraud or identity theft.

a. *Sybil attacks*: In this attack vehicles broadcast several messages containing different fake identities and locations over a certain area of interest. Then, fake traffic conditions are reported regarding that area of interest. This should affect the traffic efficiency controlling protocols by recommending vehicles to avoid the fake congested area or reschedule the located traffic lights to reduce that fake congestion.

b. *Masquerading*: Some attackers use fake identity that is related to other vehicles or RSUs. These attackers aim at utilizing some facilities and functionalities through legitimate access identification. This can be achieved by spoofing the identity of other nodes or replaying some legal packets (*i.e.*, man-in-the-middle attack).

c. *Non-repudiation*: Some vehicles deny sending or receiving a certain packet over the network. In this case, senders can send a damage data without being asked to take responsibility of sending such data. Moreover, any vehicle can deny receiving some vital packets that it did not obey and then it has caused a chaos on the road network.

3) **Privacy**: Deals with the ability a driver has to determine what data to be shared with third parties. Moreover, if the driver has to reveal his/her identity when sending a message or it can be sent anonymously.

a. *Traceability*: This threat defines the ability of tracing a certain vehicle actions over the network. This includes broadcast messages, request services or reporting cases. Tracing the actions of vehicles on the road network helps to trace their locations and identity.

b. *Linkability*: This refers to the case that an unauthorized entity can link a vehicle identity to its driver/owner. This is introduced for localizing people and tracing their information.

4 The Proposed Secure Traffic Efficiency Control Protocol: STEP

This section presents the details of the proposed Secure Traffic Efficiency control Protocol (STEP). As discussed in Section 2 several protocols were proposed to control traffic efficiency over the road network in the downtown areas. In those protocols, transmitting packets among traveling vehicles (V2V) and transmitting packets between vehicles and installed Road-Side-Units (V2I) have been used to provide real-time and efficient recommendations. Several RSUs are expected to be installed over downtown areas that can help to strength the communications as a backbone to all real-time protocols. In order to secure the traffic efficiency control applications over the downtown areas we propose the STEP protocol.

4.1 Basic Setup of STEP

The STEP protocol provides secure communications among travelling vehicles over downtown areas using the group-based cryptography technique. Vehicles transfer encrypted messages that only targeted receiver/receivers can understand, without the need of revealing the identity or the privacy of any vehicle. In traffic efficiency protocols, each vehicle is interested to transmit messages toward its neighboring vehicles (*i.e.*, same road segment in the downtown). In this work we define the road segment over downtown areas as the road between two adjacent road intersections. Thus, several road segments are configured geographically close each other in order to enhance the management processes there.

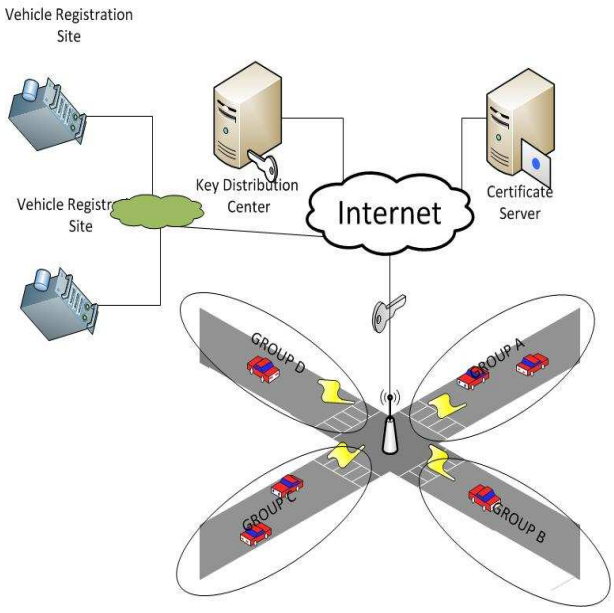


Figure 1: Downtown STEP authentication scenario

The installed RSUs over downtown areas are connected

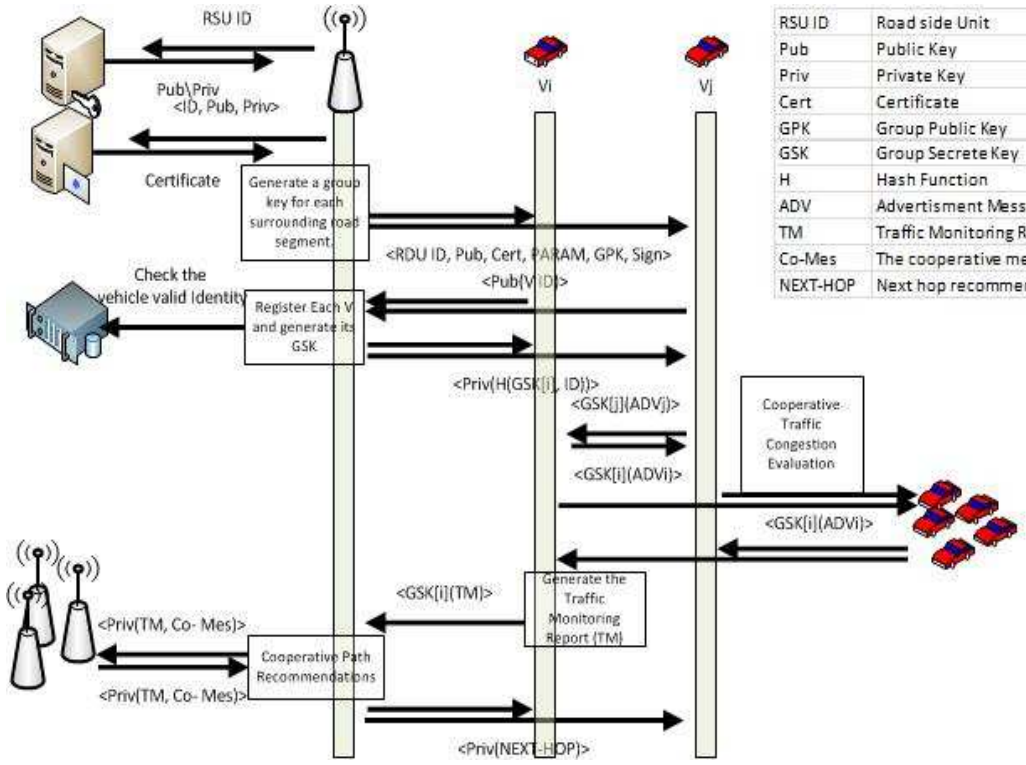


Figure 2: Phases of STEP

to the country vehicles registration authorities. As illustrated in Figure 1 several groups are configured over there. RSUs are responsible of providing each vehicle with the required variables to generate its key at the configured group. Several groups are handled by the same RSU over the road networks, each RSU should be able to prove its identity into vehicles aiming to achieve integrity, authenticity and privacy of communications.

RSUs provide the certificate authorities of vehicles since it is directly connected to vehicles registration authority. Each RSU provides close vehicles with the required variables to generate the group-key at each configured road segment, we assume that each group should be on the same road segment. Each RSU handles several road segments (*i.e.*, several groups) over the downtown scenarios. However, the RSU should be able to prove its identity in order to guarantee the integrity and authenticity of its communications. Each RSU contacts the *Key Distribution Center* to obtain public and private key pair, (Pub_i, $Priv_i$). Moreover, the *Certificate Server* provides the RSU with a certificate that contains: RSU's identity and public key that is encrypted by the Certificate Server's private key. This certificate has an expiration time and they are timestamped to prevent replay attacks. Figure 1 illustrates how RSUs contact with *Key Distribution Center* and *Certificate Server* over downtown areas.

RSUs generate the required bilinear groups with the following road segment parameters: Let $Group_1$ and $Group_2$ be two multiplicative cyclic groups of the same prime order p, gen_1 and gen_2 are generators of $Group_1$ and $Group_2$ respectively. The computable map with the Bilinearity and Nondegeneracy properties is represented by the following relation $e : Group_1 \times Group_2 \rightarrow Group_T$ [19]. ψ is a computable isomorphism from $Group_1$ to $Group_2$, with $\psi(gen_1) = gen_2$. Then, each RSU selects two random elements r and r_0, where $r \in Group_1$ and $r \neq 1_{Group_1}$, $r_0 \in Group_2$ and $r_0 \neq 1_{Group_2}$. That RSU also selects two random numbers $\xi_1, \xi_2 \in Z_q^*$, and sets $u, v \in Group_1$ such that $u^{\xi_1} = v^{\xi_2} = r$ and $r_1, r_2 \in Group_2$ such that $r_1 = {r_0}^{\xi_1}$, $r_2 = {r_0}^{\xi_2}$. The RSU randomly selects $\gamma \in Z_q^*$ as a private key and sets $w = {gen_2}^{\gamma}$ as a system parameter. A secure hash function, $Hash$, is randomly chosen for each road segment too.

The system parameters (PR) after these computations are represented by: $Group_1$, $Group_2$, $Group_T$, gen_1, gen_2, p, ψ, e, $Hash$, w, u, v, r, r_0, r_1, and r_2. The group public key (GPK) is represented by the following parameters: gen_1, gen_2, w. We assume that the Strong Diffie-Hellman (SDH) assumptions hold on $Group_1 and Groip_2$ [6] and the linear Diffie-Hellman assumption hold on $Group_1$ [8].

Each RSU broadcasts an initialization message, I_{messg}, the latter message contains that RSU's *ID, public key and certificate*. It also contains the targeted road segment's *ID*, the system parameters (PR) and group public key GPK. In order to guarantee the integrity, each RSU uses its private key to encrypt the road segment's *ID, PR and GPK* and adds the encrypted data (Enc_{data}) to the I_{messg} message.

Upon receiving any I_{messg}, any traveling vehicle, V_i, first uses the Certificate Server public key to verify the identity and the public key of that RSU. Then, it uses the RSU's public key to verify the integrity of the intended road segment identity and the generated group key. Only if V_i is currently located on the same road segment, it keeps the values of PR and GPK in its database. Consequently, the vehicle, V_i, sends a request message to the RSU aiming to register to that group, where the requested message includes an encrypted value of the real identity ID_i of V_i, using the RSU's public key. The RSU generates a private key $GSK[i]$ for each V_i with its identity. The $GSK[i]$ is represented by (A_i, x_i), where $x_i \leftarrow H(\gamma, ID_i) \in Z_q^*$ and $A_i \leftarrow {g_1}^{1/(\gamma+x_i)}$. It stores (A_i, ID_i) in its database, aiming to guarantee conditional privacy. Then, the RSU uses the secure hash function to encrypt the secrete key of that vehicle, $H(GSK[i], ID_i)$. Finally, it encrypts the hashed value using the RSU's private key and sends it back to the vehicle V_i.

Thus, all RSUs and vehicles obtain their public, group and private keys. Figure 2 illustrates, in details, the sequential steps of the setup phase in a systematic manner.

4.2 Secure Traffic Data Gathering

Each traveling vehicle V_i uses its group key GSK to encrypt the advertisement message, ADV_i. This message is periodically broadcasted to announce the basic traffic data of each vehicle (*i.e.*, ID, location, speed, direction, destination, etc). On the other hand, each vehicle gathers these ADV messages from its neighboring vehicles at the same road segment to predict the traffic situation over that road segment. Vehicles use the group public key, GPK to retrieve the guaranteed basic traffic data of the sender vehicle. In the case any suspicious message is received it can be simply dropped. Only messages that satisfy the security requirements (*i.e.*, retrieve correct data after decrypting by GPK) can be used to evaluate the traffic characteristics on the road network.

The vehicle located in the closest location to the corresponding RSU, V_C, uses the gathered traffic data to generate the traffic monitoring report of that road segment. This report includes: Traffic speed (*i.e.*, average speed of all vehicles), traffic density (*i.e.*, number of vehicles per meter square) and the expected traveling time of that road segment (based of the road segment length and the traffic speed there). V_C sends the traffic monitoring report encrypted by its GSK key. The receiver RSU uses the GPK to verify the identity of the sender vehicle and the correctness of the received data [8].

4.3 Secure Efficient Path Recommendation

Based on the traffic distribution over the road network, the most efficient path toward any targeted destination is configured at each installed RSU. In the case that, vehicles contact the located RSUs with its targeted destinations to request the best path toward their destinations. The requested message is encrypted by the GSK in oder to secure the targeted destination of the vehicle and to guarantee the authenticity. The located RSU uses GPK to read the details of the message. Then, it replies with the best path recommendation message that is encrypted by GPK.

On other cases, when the RSU periodically broadcasts a list of common targeted destinations and the next hop toward each one. In this scenario, the RSU should add a digital signature to the broadcast message to prove the integrity and authenticity of the message. Thus, malicious nodes cannot impersonate RSUs and direct the vehicles falsely.

4.4 Secure Traffic Light Controlling

Intelligent traffic lights are located as RSUs at the road intersections. Each traffic light is provided with wireless transceiver and simple processor. Traffic lights aim to guarantee safe sharing of the road intersections where conflicted flows of traffic can pass the road intersection. The schedule of each intelligent traffic light is set based on the real-time traffic characteristics of the competing flows of traffic. The sequences and the assigned time of each phase are set to minimize the queuing delay time and to increase the throughput of the signalized road intersection.

The traffic characteristics of each flow of traffic that are delivered to the scheduling processor should be encrypted using GSK of the reported vehicle. The receiver processor (RSU) uses the GPK key to verify the correctness of the received data and to verify the identity of the sender. Moreover, the schedule of each traffic light should be encrypted using the private key of the RSU. The receiver vehicles use the public key of the RSU to verify the correctness of the received data. It also checks the identity of the RSU in order to check any fake announced schedule.

5 Performance Evaluation

This section investigates the benefits of the proposed protocol in terms of controlling the traffic efficiency over the road network. This study takes its place in the case that different malicious nodes are existed and transfer fake data aiming to deceive drivers. We compare the performance of the STEP protocol to different traffic controlling protocols, where different number of malicious vehicles were detected.

5.1 Accuracy of Data Gathering

Here, we evaluate the accuracy of the gathered traffic data over the road network. This data is used to control the traffic efficiently. Malicious vehicles broadcast several advertisement messages with different identities and fake basic data. We compare the performance of the STEP protocol to ECODE [34](*i.e.*, one of the traffic evaluation

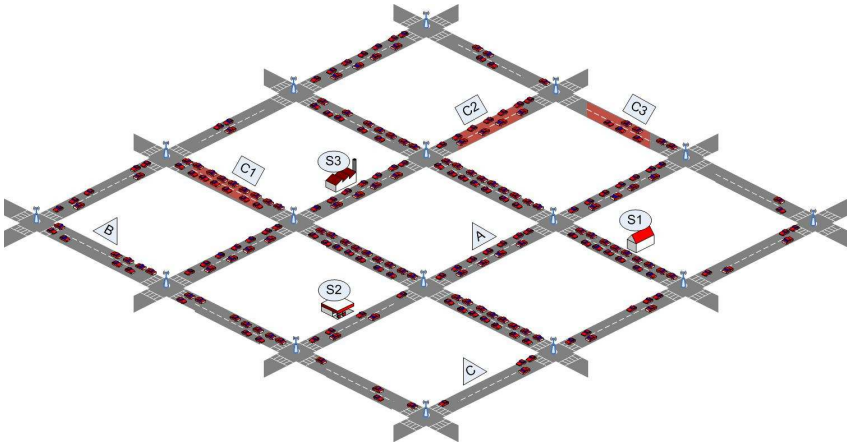


Figure 3: An example of 4X4 manhattan and three targeted destinations (A, B and C)

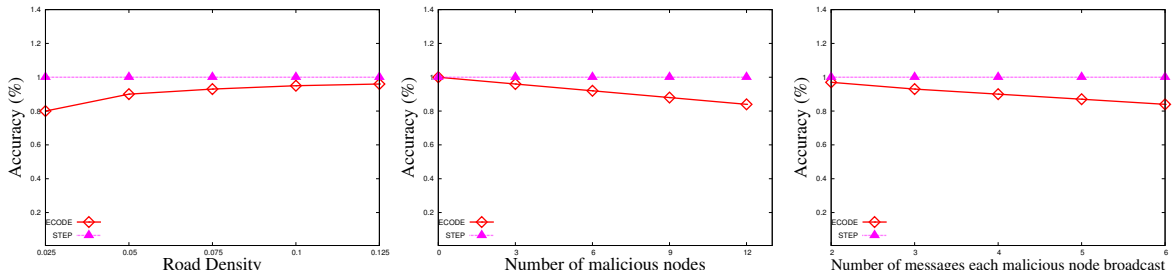


Figure 4: Data gathering accuracy of STEP: (a) Accuracy of STEP compared to ECODE for different traffic densities, (b) Accuracy of STEP compared to ECODE for different number of malicious nodes and (c) Accuracy of STEP compared to ECODE when each malicious node send different number of advertisement messages

protocols for road networks) in term of the accuracy of data gathering. We measure the accuracy of each protocol by comparing the number of detected vehicles to the number of existed vehicles over each road segment. Table 3 illustrates the simulation parameters of the performance comparison.

Table 1: Simulation parameters

Parameter	**Value**
Simulator	NS-2, SUMO
Transmission range (m)	250
Simulation time (s)	2000
Simulation area (m^2)	20 X 200
Number of Vehicles	20 - 100
Simulation map	2 lane road segment
Traffic speed	1-10 m/s
Mobility model	downtown mobility
Number of malicious nodes	4-12
Number of fake messages	2-5

In Figure 4, the comparative results of the data gathering accuracy for STEP and ECODE protocols are presented. First, in Figure 4(a) we assume the existence of five malicious nodes each one broadcast five advertisement messages. From this figure we can infer that the importance of securing the data gathering protocols is increased when the traffic density is less over the road network. By increasing the traffic density while the same number of malicious nodes are existed the effect of these nodes is becoming negligible regarding the traffic evaluation.

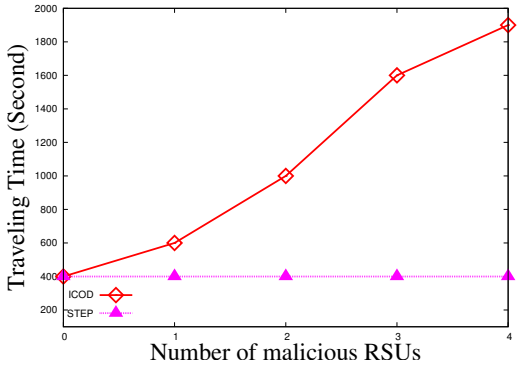
Second, we study the effect of increasing the number of malicious nodes over the road scenario where the traffic density is fixed to 0.075 $vehicle/meter^2$. The results of comparison is illustrated in Figure 4(b), several malicious nodes are simulated where each node broadcast five advertisement messages. From Figure 4(b) we can clearly see that by increasing the number of the malicious nodes the accuracy of traffic evaluation is decreased without using the secure protocol.

Figure 4(c) investigates the effect of the number of advertisement messages that each malicious node send. The malicious node becomes more disturbing when it broadcasts more fake messages in ECODE. Figure 4(a), Figure 4(b) and Figure 4(c) have shown that the STEP protocol can detect all fake messages broadcast by malicious nodes. Thus, it is able to produce accurate traffic evaluation.

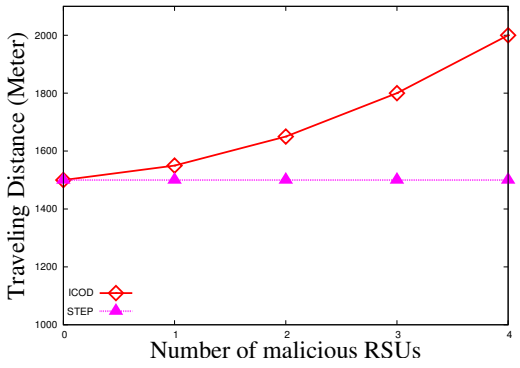
Table 2: Simulation parameters

Parameter	Value
Simulator	NS-2, SUMO
Transmission range (m)	250
Simulation time (s)	2000
Simulation area (m^2)	1000 X 1000
Number of vehicles	200 - 1000
Simulation map	Grid-layout
Mobility model	downtown mobility
Traffic speed	1-10 m/s
Number of malicious RSUs	0-4

5.2 Efficiency of The Configured Path



(a)



(b)

Figure 5: Efficiency of STEP compared to ICOD: (a) Traveling time of the configured path, (b) Traveling distance of the configured path

In this section we investigate the effects of the existence of some malicious RSUs on the road network in terms of configuring the efficient path toward targeted destinations. We run our experiments in 4X4 Manhattan model 16 RSUs are expected to be installed one at each intersection. We assume that all drivers on this road network are targeting one of three destinations, A, B and C, as illustrated in Figure 3. Some road segments are highly congested while others witness a low traffic density. Malicious RSUs broadcast fake and in-accurate data about one of the targeted destinations, drivers will be deceived to travel more time and extra distance then. Table 2 illustrates the simulation parameters for this comparison experiment.

We compare the performance of the STEP protocol in terms of configuring efficient path to ICOD [38] one of the distributed path recommendation protocols. Figure 5 illustrates the comparison study between these protocols. In Figure 5(a) we can see that by increasing number of malicious RSUs over the road network, the average traveling time toward these destinations is increased drastically by ICOD. This is due to recommending the highly congested road segments on the road network in these cases. At the same time Figure 5(b), the average traveling distance is increased when using ICOD to configure the efficient paths. However the increased in the traveling distance is small compared to the increase in the traveling time, this can be clearly seen from Figure 5(a) and Figure 5(b). The STEP protocol was able to configure malicious RSUs and ignore the recommendation messages they broadcast. Thus, the STEP protocol acquire better traveling time and traveling distance regardless of the number of existed malicious RSUs.

5.3 Efficiency of The Traffic Light Schedule

Table 3: Simulation parameters of ITLC

Parameter	Value
Simulator	NS-2 , SUMO
Transmission range (m)	250
Simulation time (s)	2000
Simulation area (m^2)	1000 X 1000
Number of traffic lights	1
Number of vehicles	200 - 1000
Simulation map	4-leg traffic intersection
Mobility model	downtown mobility
Number of malicious RSUs	0-4

We measure the efficiency of the traffic light schedule by the average waiting delay time of each vehicle at the traffic light and the throughput of the signalized road intersection (*i.e.*, number of vehicles passing the intersection per second). Malicious drivers can deceive the traffic light by broadcasting several advertisement messages to increase the traffic density of the traffic flow. Moreover, they can announce themselves as emergency vehicles that have a higher priority to pass the signalized intersection first. This drastically decrease the efficiency performance of the traffic light schedule. Figure 6 compares the STEP protocol to ITLC [36] in term of efficiency of the traffic light schedule. The average waiting delay of each vehicle is illustrated in Figure 6(a). As we can see from this figure by increasing the number of malicious drivers at the signalized intersection, the waiting delay time of each vehicle is increased when using the ITLC protocol. On the other hand, Figure 6(b) shows that using ITLC protocol to gen-

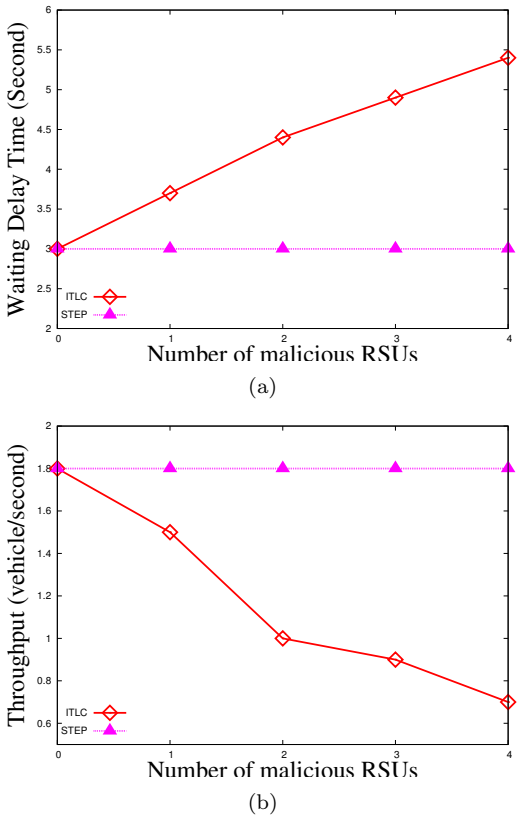


Figure 6: The efficiency of traffic light schedule: (a) Average waiting delay of each vehicle, (b) Throughput of the signalized intersection

erate the schedule of the traffic light, the throughput of the signalized intersection is decreased.

6 Conclusions

In this paper, we have proposed a secure traffic congestion control protocol, STEP. This protocol controls the traffic congestion problem over the downtown areas in a secure and efficient fashion. It relays on the public cryptography to authenticate RSUs at road intersections. On the other hand, at each road segment the group signature is used to secure the communications between vehicles. Experimental results have indicated that the efficiency protocols achieves better performance in the case that secure communications are used. This is due to the behavior of the malicious nodes which intend to deceive the efficiency control protocols aiming to serve their benefits.

References

[1] K. Abrougui and A. Boukerche, "An efficient secure service discovery protocol for intelligent transportation systems," in *IEEE 22nd International Symposium on Personal Indoor and Mobile Radio Communications (PIMRC'11)*, pp. 756–760, 2011.

[2] K. Abrougui, A. Boukerche, and Y. Wang, "Secure gateway localization and communication system for vehicular ad hoc networks," in *IEEE Global Communications Conference (GLOBECOM'12)*, pp. 391–396, 2012.

[3] O. Abumansoor and A. Boukerche, "A secure cooperative approach for nonline-of-sight location verification in vanet," *IEEE Transactions on Vehicular Technology*, vol. 61, no. 1, pp. 275–285, 2012.

[4] Q. A. Arain, D. Zhongliang, I. Memon, S. Arain, F. K. Shaikh, A. Zubedi, M. A. Unar, A. Ashraf, and R. Shaikh, "Privacy preserving dynamic pseudonym-based multiple mix-zones authentication protocol over road networks," *Wireless Personal Communications*, vol. 95, no. 2, pp. 505–521, 2017.

[5] C. T. Barba, M. A. Mateos, P. R. Soto, A. M. Mezher, and M. A. Igartua, "Smart city for vanets using warning messages, traffic statistics and intelligent traffic lights," in *IEEE Intelligent Vehicles Symposium (IV'12)*, pp. 902–907 , 2012.

[6] N. Barić and B. Pfitzmann, "Collision-free accumulators and fail-stop signature schemes without trees," in *International Conference on the Theory and Applications of Cryptographic Techniques*, pp. 480–494, 1997.

[7] P. Bellavista, F. Caselli, A. Corradi, and L. Foschini, "Cooperative vehicular traffic monitoring in realistic low penetration scenarios: The colombo experience," *Sensors*, vol. 18, no. 3, pp. 822, 2018.

[8] D. Boneh, X. Boyen, and H. Shacham, "Short group signatures," in *Annual International Cryptology Conference*, pp. 41–55, 2004.

[9] W. Chen and Y. Sun, "On the security cost of interval multicast," in *International Conference on Information and Automation (ICIA'09)*, pp. 101–105 , 2009.

[10] M. S. Hwang C. Y. Tsai, P. F. Ho, "A secure group signature scheme," *International Journal of Network Security*, vol. 20, no. 2, pp. 201–205, 2018.

[11] L. D. Chou, D. C. Li, and H. W. Chao, "Mitigate traffic congestion with virtual data sink based information dissemination in intelligent transportation system," in *Third International Conference on Ubiquitous and Future Networks (ICUFN'11)*, pp. 37–42 , 2011.

[12] E. C. Eze, S. J. Zhang, E. J. Liu, and J. C. Eze, "Advances in vehicular ad-hoc networks (VANETs): Challenges and road-map for future development," *International Journal of Automation and Computing*, vol. 13, no. 1, pp. 1–18, 2016.

[13] J. Fukumoto, N. Sirokane, Y. Ishikawa, T. Wada, K. Ohtsuki, and H. Okada, "Analytic method for real-time traffic problems by using contents oriented communications in vanet," in *7th International Conference on ITS Telecommunications (ITST'07)*, pp. 1–6 , 2007.

[14] R. L. Gordon, W. Tighe, ITS Siemens, *et al.*, Traffic Control Systems Handbook, FHWA-HOP-06-006, 2005.

[15] V. Gradinescu, C. Gorgorin, R. Diaconescu, V. Cristea, and L. Iftode, "Adaptive traffic lights using car-to-car communication," in *65th Vehicular Technology Conference (VTC'07)*, pp. 21–25, 2007.

[16] D. Greenwood, B. Burdiliak, I. Trencansky, H. Armbruster, and C. Dannegger, "Greenwave distributed traffic intersection control," in *Proceedings of The 8th International Conference on Autonomous Agents and Multiagent Systems-Volume 2*, pp. 1413–1414, 2009.

[17] W. J. Hu, Y. B. Huang, Q. Y. Zhang, "Robust speech perception hashing authentication algorithm based on spectral subtraction and multi-feature tensor," *International Journal of Network Security*, vol. 20, no. 2, pp. 206–216, 2018.

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

[19] X. Lin, X. Sun, P. H. Ho, and X. Shen, "Gsis: A secure and privacy-preserving protocol for vehicular communications," *IEEE Transactions on Vehicular Technology*, vol. 56, no. 6, pp. 3442–3456, 2007.

[20] H. Menouar, I. Guvenc, K. Akkaya, A. S. Uluagac, A. Kadri, and A. Tuncer, "Uav-enabled intelligent transportation systems for the smart city: Applications and challenges," *IEEE Communications Magazine*, vol. 55, no. 3, pp. 22–28, 2017.

[21] K. Nellore and G. P. Hancke, "A survey on urban traffic management system using wireless sensor networks," *Sensors*, vol. 16, no. 2, pp. 157, 2016.

[22] Z. Ozcelik, C. Tastimur, M. Karakose, and E. Akin, "A vision based traffic light detection and recognition approach for intelligent vehicles," in *International Conference on Computer Science and Engineering (UBMK'17)*, pp. 424–429 , 2017.

[23] K. Pandit, D. Ghosal, H. M. Zhang, and C. N. Chuah, "Adaptive traffic signal control with vehicular ad hoc networks," *IEEE Transactions on Vehicular Technology*, vol. 62, no. 4, pp. 1459–1471, 2013.

[24] K. Pandit, D. Ghosal, H. M. Zhang, and C. N. Chuah, "Adaptive traffic signal control with vehicular ad hoc networks," *IEEE Transactions on Vehicular Technology*, vol. 62, no. 4, pp. 1459–1471, 2013.

[25] J. Qian and R. Eglese, "Fuel emissions optimization in vehicle routing problems with time-varying speeds," *European Journal of Operational Research*, vol. 248, no. 3, pp. 840–848, 2016.

[26] M. Riley, K. Akkaya, and K. Fong, "Group-based hybrid authentication scheme for cooperative collision warnings in vanets," *Security and Communication Networks*, vol. 4, no. 12, pp. 1469–1482, 2011.

[27] M. A. Salman, S. Ozdemir, and F. V. Celebi, "Fuzzy logic based traffic surveillance system using cooperative v2x protocols with low penetration rate," in *International Symposium on Networks, Computers and Communications (ISNCC'17)*, pp. 1–6 , 2017.

[28] M. Sankaranarayanan, C. Mala, and S. Mathew, "Congestion rate estimation for vanet infrastructure using fuzzy logic," in *Proceedings of the International Conference on Intelligent Systems, Metaheuristics & Swarm Intelligence*, pp. 98–102, 2017.

[29] O. Tomescu, I. M. Moise, A. E. Stanciu, and I. Batros, "Adaptive traffic light control system using ad hoc vehicular communications network," *UPB Scientific Bulletin, Series D*, vol. 74, no. 2, 2012.

[30] K. Upasani, M. Bakshi, V. Pandhare, and B. K. Lad, "Distributed maintenance planning in manufacturing industries," *Computers & Industrial Engineering*, vol. 108, pp. 1–14, 2017.

[31] C. Vilarinho, J. P. Tavares, and R. J. F. Rossetti, "Intelligent traffic lights: Green time period negotiaton," *Transportation Research Procedia*, vol. 22, pp. 325–334, 2017.

[32] M. B. Younes, G. R. Alonso, and A. Boukerche, "A distributed infrastructure-based congestion avoidance protocol for vehicular ad hoc networks," in *IEEE Global Communications Conference (GLOBECOM'12)*, pp. 73–78 , 2012.

[33] M. B. Younes and A. Boukerche, "A performance evaluation of a context-aware path recommendation protocol for vehicular ad-hoc networks," in *IEEE Global Communications Conference (GLOBECOM'13)*, pp. 516–521 , 2013.

[34] M. B. Younes and A. Boukerche, "A performance evaluation of an efficient traffic congestion detection protocol (ECODE) for intelligent transportation systems," *Ad Hoc Networks*, vol. 24, pp. 317–336, 2015.

[35] M. B. Younes and A. Boukerche, "Scool: A secure traffic congestion control protocol for vanets," in *IEEE Wireless Communications and Networking Conference (WCNC'15)*, pp. 1960–1965 , 2015.

[36] M. B. Younes and A. Boukerche, "Intelligent traffic light controlling algorithms using vehicular networks," *IEEE Transactions on Vehicular Technology*, vol. 65, no. 8, pp. 5887–5899, 2016.

[37] M. B. Younes and A. Boukerche, "An efficient dynamic traffic light scheduling algorithm considering emergency vehicles for intelligent transportation systems," *Wireless Networks*, pp. 1–13, 2017.

[38] M. B. Younes, A. Boukerche, and G. Rom'an-Alonso, "An intelligent path recommendation protocol (icod) for vanets," *Computer Networks*, vol. 64, pp. 225–242, 2014.

[39] M. B. Younes, A. Boukerche, and X. Zhou, "An intelligent vehicular traffic prediction (itp) protocol," in *IEEE 40th Local Computer Networks Conference Workshops (LCN Workshops)*, pp. 899–904, 2015.

[40] X. Zhang, V. J. Karplus, T. Qi, D. Zhang, and J. He, "Carbon emissions in china: How far can new efforts bend the curve?," *Energy Economics*, vol. 54, pp. 388–395, 2016.

[41] J. Zhong and H. Liao, "Research on the applications of electronic information technology in intelligent

traffic light signal control," in *Proceedings of International Conference on Computing,Communications and Automation (I3CA'17)*, 2017.

[42] S. Zhong, X. Xiao, M. Bushell, and H. Sun, "Optimal road congestion pricing for both traffic efficiency and safety under demand uncertainty," *Journal of Transportation Engineering, Part A: Systems*, vol. 143, no. 4, pp. 04017004, 2017.

Biography

Maram Bani Younes received the Ph. D. degree in computer science from the University of Ottawa, Ottawa, ON, Canada, in 2015. She is currently an Assistant Professor with the Department of Computer Science, Philadelphia University, Jordan, and a Research Associate with the School of Information Technology and Engineering, University of Ottawa. Her research interests are wireless networks, wireless ad hoc and sensor networks, vehicular networks and traffic efficiency for vehicular network.

A New Erasure Code Decoding Algorithm

Di Fan, Feng Xiao, and Dan Tang

(Corresponding author: Dan Tang)

SoftwareEngineering College, Chengdu University of Information Technology
No. 24, Xue-Fu Road, Chengdu 610225, China
(Email: tangdan@foxmail.com)

(Received Nov. 23, 2017; Revised and Accepted June 18, 2018; First Online Feb. 24, 2019)

Abstract

Erasure code, as a fault-tolerant technology which is widely used in storage and communication fields. It can reduce the storage space consumption and provide the fault-tolerant capability of the replication backup. Therefore, a simple and efficient erasure coding and decoding algorithm is also paid much attention. Deenadhayalan has proposed a matrix methods for lost data reconstruction for erasure code that uses the pseudo-inverse principle to recover a random raw data element and apply to any erasure code, but cannot recover both the data element and the redundant element at the same time. After the data elements are recovered, the redundant elements can recovered by encoding, which increases the computational complexity. In response to this situation, this paper presents an improved decoding algorithm suitable for any theoretically recoverable cases. The algorithm is an erasure code decoding algorithm based on matrix that can reconstruct data elements and redundant elements at the same time. It also has high practicability and simple,easy algorithm steps, is easy to implement and has a wide range of applications. Through the simulation experiment it can be concluded that the efficiency is also high.

Keywords: Erasure Codes; Improved Decoding Algorithm; Matrix Decoding; Theory Can be Restored

1 Introduction

With the rapid development of computer technology, information technology has been widely popularized in various industries and fields. The data were explosively growing, and the demand for the storage system [6, 12, 13] is getting higher and higher. With the increasing storage demand, both the number of storage nodes and the capacity of single node in storage system are increasing exponentially, which means that the probability of node failure and the failure of sectors in single node are larger than before, so data fault-tolerant is an indispensable key technology in storage system.

The most widely used fault tolerance technology is multi copy replication technology, that is, fault-tolerance by replica copy. The other is erasure code technology, through the encoding of fault-tolerance. Erasure code technology [7, 10] mainly relies on the erasure code algorithm [14] to store the original data after obtaining redundant elements, so as to achieve the purpose of fault tolerance. In the storage system, its main idea is to encode the original data element of the k block to obtain the m block redundancy element, and when there is a m block element failures, the lost element can be recovered through the remaining elements by using a certain decoding algorithm. Compared with multi-rseplica fault-tolerant technology, erasure code fault-tolerant technology can reduce the storage space significantly while providing the same or even higher data fault tolerance.

In recent years, most of the research on erasure code is focused on the encoding process [8], and the decoding process is rarely involved. The decoding process of the original erasure code is processed by cyclic iteration or matrix inversion. Each code has different decoding algorithm. And the original decoding type is node loss, when an element or sector is lost in a node, the entire node is considered invalid. However, as the amount of data is increasing and the number of hardware is increasing, there are more and more failures of sectors in one node. When the whole node is rebuilt, those sectors that are not needed to be rebuilt are also rebuilt, thus causing repetition and increasing unnecessary computation. Therefore, the restoration of random elements or sector losses has also become an important problem in erasure code decoding.

In [9], an algorithm for merging and decoding in binary domain (hereinafter referred to as merger decoding) is proposed. This algorithm restores the node error by rebuilding the data block on the fault-tolerant storage system and and can be used to restore the loss of the random element. However, the calculation of this algorithm involves the calculation of the inverse matrix. Therefore, when the single error is restored, the efficiency will be higher. Once there are many errors, the operation of inversion will greatly affect the speed of operation, thus affecting the decoding efficiency.

In [3], Deenadhayalan proposed a decoding algorithm for erasure codes. This algorithm is based on generator matrix and its pseudo-inverse matrix (hereinafter referred to as matrix decoding), and is generally declared as two results for lost data sectors. One is that the algorithm is recoverable, which is theoretically recoverable, when the algorithm provides a formula made up of readable data to restore the lost sector, the other is an unrecoverable sector in theory. The matrix decoding algorithm not only solves the problem of recovery of random sector loss, but also abandons the calculation of the inverse matrix to make it highly efficient. At the same time, it is also a universal decoding algorithm that is applicable to any array code and can also be used for non-XOR erasure code, but most suitable for the array code. Therefore, this paper describes the array code as an example. However, the matrix decoding algorithm has a disadvantage at the same time. The loss of redundant elements can only be solved by the encoding algorithm after the data elements are recovered, but cannot recover the data elements and redundant elements at the same time. In this paper, an improved algorithm which can restore random lost sector including redundant sector is proposed for the problem of matrix decoding. It reduces the complexity of algorithm from the algorithm level, and can restore any loss that can be recovered theoretically, and the efficiency has been improved through experiments.

Here is a description of the organizational structure of this paper. The second part introduces some basic theories and principles implicated in the algorithm. The third part presents the example of the original matrix decoding algorithm and the improved algorithm steps, and gives concrete examples. The fourth part analyzes the experimental data of the algorithm, and compared with the other decoding algorithm by performance. The fifth part gives the summary of this paper.

2 Basic Concepts and Principles

In order to better describe the algorithm and get a clearer understanding of the algorithm, this section will introduce some basic concepts and principles involved in the paper.

2.1 Basic Concepts

There is no consistent definition of erasure-tolerant technology to erasing codes in storage systems. In order to facilitate the description and understanding of this paper, based on the literature [3, 4], the relevant concepts commonly used in this paper are as follows.

- Data (or information): The original piece of data string used to store the information needed by real users.
- Parity (or redundant): By using the erasure code algorithm, a data string of redundant information obtained by calculating the data, the existence of these redundancies is to ensure the erasure code's fault-tolerance.
- Element (or symbol): A fundamental unit of data or parity; this is the building block of the erasure code. In the process of erasure code calculation, an element is usually regarded as a basic unit of computation.
- Stripe: Collection of all information independently related to the same erasure algorithm. A storage system can be regarded as a collection of multiple stripes. The stripe is a set of information that independently constitutes an erasure code algorithm.
- Strip: A unit of storage consisting of all continguous elements (data, parity or both) from the same disk and stripe. In coding theory,this is associated with a code symbol. It is sometimes called a stripe unit.The set of strips in code instance form a stripe.Typically,the strips are all of the same size(contain the same number of elements). A collection of data belonging to the same stripe on the same disk. The size of a strip is determined by the number of elements contained in the strip.

The symbols and descriptions of some principles are described in Table 1.

Table 1: Symbols

Symbols	Size	Explain
G	$R \times C$	Generate matrix
H	$(R-C) \times R$	Check matrix
U	$C \times R$	Left pseudo-inverse
O_R	$C \times C$	Partial unit matrix
$d_{i,j}$	—	The i element of the j strip in the disk array
D	$C \times 1$	Raw data
T	$R \times 1$	After encoding the element data
H'	$(R-C) \times R$	Redundancy matrix

2.2 Basic Principles and Proofs

The basic principle of the improved algorithm in this paper is divided into two parts: the pseudo-inverse matrix U used to recover the data elements and the redundancy matrix H' used to recover the redundant elements. Then we describe separately and give the proof at the same time.

2.2.1 Pseudo-Inverse Matrix U

First describe some of the basic theories about linear algebra in binary.

Definition 1.

(Left pseudo-inverse): If matrix A is left multiplied by matrix B to get the unit matrix, then B is called the left pseudo-inverse matrix of A. When the matrix is full rank and $R \leq C$, the left pseudo-inverse matrix must exist.

(Null space): Null space refers to a set of all vectors orthogonal to each row vector of the matrix. Null Space Base refers to the largest set of linearly independent vectors in null space.

Suppose that G is a $R \times C$ matrix, and $R \leq C$.If B is the null space of G, U is the left pseudo-inverse matrix of G, X varies over all binary $C \times (R - C)$ matrices, then get Equation (1):

$$(U + (X \cdot B)) \cdot G = O_R. \tag{1}$$

$U + (X \cdot B)$ runs over all partial pseudo-inverses, X is to add a null space vector for each column of U. There are two important equations in the coding theory, $G \times D = T$ and $H \times T = 0$ respectively. This can be introduced Equation (2):

$$H \times G \times D == 0. \tag{2}$$

It can be seen from Equation (2) that H is a zero-space basis of G, so that the pseudo-inverse matrix of the generated matrix can be found using the check matrix.

Theorem 1. *The left pseudo-inverse matrix obtained by the improved algorithm, in which any theoretically recoverable data element corresponds to a non-zero row of the pseudo-inverse matrix, and the non-zero positions in these non-zero row indicate which data elements and redundant elements whose XOR is a data element. A directly readable element corresponds to a labeled row in the pseudo-inverse matrix (ie, a row vector containing only one.) An unrecoverable data element corresponds to an all-zero row of pseudo-inverse matrix.*

Proof. Suppose that T represents the vector containing all the elements after encoding, T' represents the lost coding vector, that is, the lost element corresponding position is 0, and obviously get Equation (3)

$$G' \times D = T' \tag{3}$$

□

In Equation (3), the missing element corresponds to all-zero rows in G'. So,here has Equation (4)

$$U \cdot T' = U \cdot G' \cdot D = O_R \cdot D = D' \tag{4}$$

Where the 0 element of D' corresponds to a zero position on the diagonal of O_R, and O_R corresponds to all-zero rows in the pseudo-inverse matrix. Non-zero position on the diagonal of O_R corresponds to the non-zero position in D', while the non-zero position on the diagonal of O_R corresponding to non-zero row of pseudo-inverse matrix U. Thus can be see each row of the pseudo-inverse matrix U corresponds to each of the elements of D',that is the data element. In the meantime, since $U \cdot T' = D'$, therefore, each row in the pseudo-inverse matrix U, each bit corresponds to one element in T, so that each row corresponds to one data element and each bit corresponds to one element.

2.2.2 Check Matrix H and Redundancy Matrix H'

Check matrix is a very important concept in coding theory.This section describes the concepts of check matrices,redundancy matrices and the theory of recovery parity elements. In this paper, we referred to the matrices transformed by the improved parity check matrix as redundancy matrices.

Check matrix is a matrix used to check whether a codeword is correct. Each column represents an element location. Each row represents a redundant element and is also an equation (the result is 0 after XOR for all non-zero positions in each row). For example, Equation (5) is the check matrix of STAR (3,6) code.

$$H = \begin{pmatrix} 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix} \tag{5}$$

Theorem 2. *The redundancy matrix obtained by the improved algorithm in this paper, whose non-zero rows represent a theoretically recoverable redundant element, the exclusive-OR of elements corresponding to each non-zero position in this row is the redundancy element corresponding to this row. Each of all-zero row represents a known readable redundant element.*

Proof. As described above for the check matrix, it can be clearly seen that the result of the exclusive-OR of each row in the check matrix is 0, so the result of the XOR between the row and the row is also 0, and the XOR transformations between rows and rows does not affect the property of the check matrix. Suppose that the redundant elements of STAR (3, 6) code are $(P_0, P_1|Q_{0,0}, Q_{1,0}|Q_{0,1}, Q_{1,1})$, that is, each row represents a redundant element. If the second row is added to the first row and placed in the first line, the result of XOR for all the non-zero position elements is still 0. As in Equations (6), (1) and (3) add the same result to zero.

$$\begin{cases} d_{0,0} + d_{0,1} + d_{0,2} + P_0 = 0(1) \\ d_{0,0} + d_{0,1} + d_{0,2} = P_0(2) \\ d_{1,0} + d_{1,1} + d_{1,2} + P_1 = 0(3) \\ d_{1,0} + d_{1,1} + d_{1,2} = P_1(4) \end{cases} \tag{6}$$

□

On this basis, if zero redundant element corresponding to the row, the result of the difference of the remaining elements is equal to the redundant element, so that the

redundant element can be obtained. Use the above check matrix formula for example. The Equations (6) (1) and (3) two formula will be combined make P_0 into 0, then get the available Equation (7).

$$d_{0,0} + d_{0,1} + d_{0,2} + d_{1,0} + d_{1,1} + d_{1,2} + P_1 = P_0 \quad (7)$$

From this we can prove that the theory 2 is correct.With the above concepts and theories, the next section will describe our improved algorithm process based on these contents and give examples.

3 Decoding Algorithm

In Paper [3], a matrix based erasure decoding algorithm matrix decoding is described. In the literature, specific methods and steps are given for the reconstruction of EVENODD array code [11]. For the case of missing elements including redundant elements, the matrix decoding algorithm first recovers the missing data elements and then encodes the missing redundant elements. This section first describes the decoding algorithm of the matrix decoding algorithm, and gives an example. Next, the improved algorithm of the matrix decoding algorithm in this paper is introduced, and the algorithm steps are described in detail.

3.1 Matrix Decoding Algorithm

The matrix decoding algorithm is based on the matrix theory and the pseudo-inverse principle. The core of the algorithm is to construct the pseudo-inverse matrix of the generate matrix. When the pseudo-inverse matrix is constructed, each column of the pseudo-inverse matrix represents one data element, each non-zero position of a column represents a known-readable element. H in the original algorithm is a vertical matrix. So let's describe the structure of the pseudo-inverse matrix.

1) Construct a square matrix W of size $R \times R$, $W = (B|H)$,the initial B consists of a unit matrix and all-zero rows.Write all missing element positions to a uniform list L - lost list.

2) For each lost element in list L,let r indicate the lost element corresponding to the row of W, then:

   - Find any column b in H that has a one in row r. If none exists,Zero any column in B that has a one in row r and continue to the next lost element;
   - For each one in row r of W, say in column c,if $c \neq b$, sum and replace column b into column c.
   - Zero column b in H.

3) Use the resulting B to recover lost data elements.

Example 1. *The following uses STAR(3,6) as an example. The data encoded by the STAR code is arranged as Equation (8)*

$$T = (d_{0,0}, d_{1,0} | d_{0,1}, d_{1,1} | d_{0,2}, d_{1,2} | P_0, P_1 | Q_{0,0}, Q_{1,0} | Q_{0,1}, Q_{1,1}) \quad (8)$$

Suppose that the lost elements list L = (0, 2, 4, 5, 8, 9), then the data is arranged as T = (0, $d_{1,0}$|0, $d_{1,1}$|0, 0|P_0, P_1|0, 0|$Q_{0,1}$, $Q_{1,1}$). From the steps above can get the pseudo-inverse matrix as Equation (9).

$$U = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 \end{pmatrix} \quad (9)$$

Each column in the pseudo-inverse matrix represents a primitive data element, and each non-zero position represents a known available data element. After the original data elements 0, 2, 4, and 5 are obtained therefrom, an encoding algorithm get 8,9 and multiply the original data element by the generator matrix, as in Equation (10).

$$\begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 & 1 & 0 \\ 1 & 0 & 1 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 & 1 & 1 \end{pmatrix} \begin{pmatrix} d_{0,0} \\ d_{1,0} \\ d_{0,1} \\ d_{1,1} \\ d_{0,2} \\ d_{1,2} \end{pmatrix} = \begin{pmatrix} d_{0,0} \\ d_{1,0} \\ d_{0,1} \\ d_{1,1} \\ d_{0,2} \\ d_{1,2} \\ P_0 \\ P_1 \\ Q_{0,0} \\ Q_{1,0} \\ Q_{0,1} \\ Q_{1,1} \end{pmatrix} \quad (10)$$

3.2 Improved Decoding Algorithm

This section will describe the improved algorithm proposed in this paper, will give the algorithm specific steps and examples. The algorithm proposed in this paper is based on the improvement of the matrix decoding algorithm. But it can recover all the theoretically recoverable cases at the same time, including recover the original data elements and redundant data elements at the same time. It can be applied to any array code, and it can be extended to non-binary erasure codes, such as RS codes.

3.2.1 Improved Algorithm Steps

Algorithm steps are as follows:

1) Construct a square matrix A, $A = \binom{U}{H}$, $U = (I_c|0)$, H is check matrix. Is the lost elements list;

2) Judging whether the right half of the check matrix formed is a unit matrix or not, if not, transform matrix between rows and rows to make it a identify matrix;

3) The transformation of A is equivalent to the inversion process;

4) Get the converted A can recover the lost elements, a row represents a data element, in the row a non-zero position corresponding to a known data elements.

The following steps for the transformation of the specific steps:

1) For each data element s in the lost element list L, first determine whether the type of the data element belongs to the original data or to the redundant data. If s belongs to original data,then continue; if not,then skip. Loop through the data block element s in L;

2) Find h in H, its s column is 1. If none exist, the U in the s column has a line set to zero;

3) After found h, if L does not contain redundant elements, then select the most sparse row f from the found result h, if the redundant elements are included, remove the missing redundant elements from the found result after select the most sparse row f in h (in order to retain the value of the missing redundant element row, the last can be found at the same time);

4) For one in column s of A in row e, if e is not equal to f, add f(exclusive-or) to e and replace e;

5) Set the row f in H to zero;

6) After traversing the data block elements in L, set the redundant elements in L correspond to the columns in H to zero.

3.2.2 Examples of Improved Algorithm

In order to better understand and explain the above algorithm, the following is illustrated by taking STAR [5](3,6) and RDP [2](3,4) as examples.

Eg1. STAR code: The structure of the STAR code is shown in Table 2.

Table 2: STAR code structure

S_0	S_1	S_2	P	Q_0	Q_1
$d_{0,0}$	$d_{0,1}$	$d_{0,2}$	P_0	$Q_{0,0}$	$Q_{0,1}$
$d_{1,0}$	$d_{1,1}$	$d_{1,2}$	P_1	$Q_{1,0}$	$Q_{1,1}$

Expand the disk data, change to $D = (d_{0,0}, d_{1,0}|d_{0,1}, d_{1,1}|d_{0,2}, d_{1,2})$, then $T = (d_{0,0}, d_{1,0}|d_{0,1}, d_{1,1}|d_{0,2}, d_{1,2}|P_0, P_1|Q_{0,0}, Q_{1,0}|Q_{0,1}, Q_{1,1})$. Construct the square matrix as shown in Equation (11).

$$A=\left(\frac{U}{H}\right)=\begin{pmatrix}1&0&0&0&0&0&0&0&0&0&0&0\\0&1&0&0&0&0&0&0&0&0&0&0\\0&0&1&0&0&0&0&0&0&0&0&0\\0&0&0&1&0&0&0&0&0&0&0&0\\0&0&0&0&1&0&0&0&0&0&0&0\\0&0&0&0&0&1&0&0&0&0&0&0\\1&0&1&0&1&0&1&0&0&0&0&0\\0&1&0&1&0&1&0&1&0&0&0&0\\1&0&0&1&1&1&0&0&1&0&0&0\\0&1&1&1&1&0&0&0&0&1&0&0\\1&0&1&1&0&1&0&0&0&0&1&0\\0&1&1&0&1&1&0&0&0&0&0&1\end{pmatrix} \tag{11}$$

In order to facilitate the comparison with the original algorithm, we assume that the missing element is the same as the missing element in Section 3.1.Suppose that the lost elements list $L = (0, 2, 4, 5, 8, 9)$. The data in list L correspond to rows s of A.

First determine the check matrix, the right half of it is the unit matrix, then the following operation.

For column $s = 0$, find some row in H that has a one in this column,we can find $h = (6, 8, 10)$, but 8 is in list L, so we choose $h = 6$, because the row 6 is more sparse than the row 10, after select, add row 6 to row 0,8,10, then set row 6 to 0; For row $s = 2$, we can find $h = (8, 9, 11)$, but 8,9 is in list L, so we choose $h = 11$, after select,add row 11 to row 0,2,8,9, then set row 11 to 0; For column $s = 4$, find some row in H that has a one in column 4, we can find $h = (8, 10)$, but 8 is in list L,so we choose $h = 10$, after select, add row 10 to row 2,4,8, then set row 10 to 0; For column $s = 5$, we can choose $h = (7, 8, 9)$, but 8,9 is in list L,so we choose $h = 7$, after choose, add row 7 to row 0,4,5,8,9, set row 7 to 0;For row $s = 8, 9$, because 8,9 is in list L, so end the traverse, set the column 8,9 to 0.The final result becomes Equation (12):

$$A=\begin{pmatrix}0&0&0&1&0&0&1&1&0&0&0&1\\0&1&0&0&0&0&0&0&0&0&0&0\\0&1&0&1&0&0&1&0&0&0&1&1\\0&0&0&1&0&0&0&0&0&0&0&0\\0&1&0&0&0&0&1&1&0&0&1&0\\0&1&0&1&0&0&0&1&0&0&0&0\\0&0&0&0&0&0&0&0&0&0&0&0\\0&0&0&0&0&0&0&0&0&0&0&0\\0&0&0&1&0&0&0&1&0&0&1&1\\0&1&0&0&0&0&0&1&0&0&0&1\\0&0&0&0&0&0&0&0&0&0&0&0\\0&0&0&0&0&0&0&0&0&0&0&0\end{pmatrix} \tag{12}$$

A row with multiple non-zero positions in A is a theoretically solvable element, which results in Equation (13):

$$\begin{cases}d_{0,0}=d_{1,1}+P_0+P_1+Q_{1,1}\\d_{0,1}=d_{1,0}+d_{1,1}+P_0+Q_{1,0}+Q_{1,1}\\d_{0,2}=d_{1,0}+P_0+P_1+Q_{1,0}\\d_{1,2}=d_{1,0}+d_{1,1}+P_1\\Q_{0,0}=d_{1,1}+P_1+Q_{1,0}+Q_{1,1}\\Q_{0,1}=d_{1,0}+P_1+Q_{1,1}\end{cases} \tag{13}$$

Eg2. RDP code: The structure of the RDP code is shown in Table 3.

Table 3: RDP code structure

S_0	S_1	P	Q
$d_{0,0}$	$d_{0,1}$	P_0	Q_0
$d_{1,0}$	$d_{1,1}$	P_1	Q_1

Expand the disk data, change to $D = (d_{0,0}, d_{1,0}|d_{0,1}, d_{1,1})$, then $T = (d_{0,0}, d_{1,0}|d_{0,1}, d_{1,1}|P_0, P_1|Q_0, Q_1)$. Construct the square matrix as shown in Equation (14).

$$A = \left(\frac{U}{H}\right) = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 & 0 & 1 \end{pmatrix} \tag{14}$$

Suppose that the lost elements list $L = (0, 2, 4, 5)$. First determine the check matrix, the right half of it is the unit matrix, it can be clearly seen is not, so after some transformation between rows, the row 5 added to the row 6, we can make the right half of H into a unit matrix, the final transformation results is:

$$A = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 & 0 & 1 \end{pmatrix} \tag{15}$$

For column $s = 0$,find some row in H that has a one in this column, we can find $h = (4, 6)$, but 4 is in list L, so we choose $h = 6$, after select,add row 6 to row 0,8,10,then set row 6 to 0, the result matrix is:

$$A = \begin{pmatrix} 0 & 1 & 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 & 0 & 1 \end{pmatrix} \tag{16}$$

For column $s = 2$, find some row in H that has a one in this column, $h = (4, 7)$, but 4 is in list L, so choose $h = 7$, add row 7 to row 0,4, set row 7 to 0,the result matrix is :

$$A = \begin{pmatrix} 0 & 1 & 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} \tag{17}$$

A row with multiple non-zero positions in A is a theoretically solvable element, which gives the following Equation (18):

$$\begin{cases} d_{0,0} = d_{1,0} + d_{1,1} + Q_0 \\ d_{0,1} = d_{1,0} + Q_1 \\ P_0 = d_{1,1} + Q_0 + Q_1 \\ P_1 = d_{1,0} + d_{1,1} \end{cases} \tag{18}$$

4 Analysis and Discussions

For a code system, the performance of coding system is usually evaluated from the encoding rate and the utilization ratio of space. For decoding, the decoding rate, recovery efficiency, and loss type of the decoding algorithm can be used for evaluation. This section will experimentally analyze this algorithm in terms of its applicability and decoding rate. The following is encoded and simulated data loss in the Python 3 environment supposing a folder represents a disk to emulate the raid.

Array code is a code system constructed only through XOR operations, and its own decoding algorithm uses cyclic iterative decoding. When an element in a strip is lost, it is considered the loss of the entire strip or even the entire disk. The entire disk is rebuilt upon recovery, and the original decoding of each array code is different. Deenadhaylan proposes an algorithm for restoring random data elements - matrix decoding, using the pseudo-inverse theory of the generating matrix to reconstruct the data elements, which is suitable for any erasure code, but can not restore the redundant elements at the same time. Tang proposed a merger decoding algorithm in paper [9], which reconstructed disk data elements by chunking and inverting the check matrix to recover both data elements and redundant elements. But this algorithm needs to compute the inverse matrix, which increases the computational complexity and the efficiency is not high. The improved decoding algorithm proposed in this paper is based on the improvement of the matrix decoding algorithm, and can recover any theoretically possible recovery, including the simultaneous restoration of data elements and redundant elements. Table 4 compares the three decoding algorithms and the cyclic iterative decoding algorithm for the decoding of the array code in terms of whether it can restore the random elements, whether it can restore data and redundant data and versatility at the same time.

It can be seen from Table 4 that the three algorithms of matrix decoding, merge decoding and improved decoding algorithm can recover random elements under the same fault-tolerant capacity. The merge decoding and the improved decoding algorithm can also recover the data and redundant elements. Compared with the original decoding algorithm only for one array code, the improved decoding algorithm and the merger decoding algorithm can be applied to any array code.

The following is the analysis of experimental data. For the matrix decoding algorithm, when the pseudo-inverse

Table 4: Properties of algorithm

Decoding algorithm	Whether the random elements can be recovered	Recover data and redundant elements at the same time	Versatility
Cyclic iterative	False	False	Bad
Matrix decoding	True	False	Good
Merger decoding	True	True	Good
Improved decoding	True	True	Good

matrix is taking, it needs to be encoded to obtain the redundant elements. Therefore, an extra matrix operation is added to the algorithm complexity of the improved algorithm in this paper. Taking EVENODD [1] for example, the matrix decoding algorithm requires 60 more XOR operations than the algorithm proposed in this paper, each adding 60 more operations. Even if the XOR operation is fast, the efficiency will still be affected. This paper simulates the data loss and compares efficiency by decoding different file sizes. The final experimental results are shown in Figure 1.

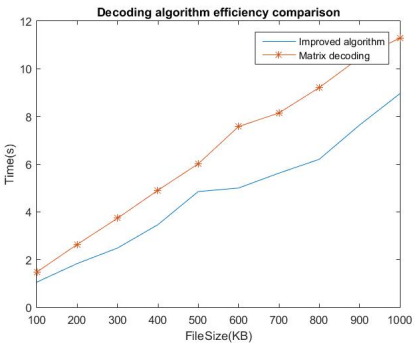


Figure 1: Efficiency comparison

It can be seen from the figure above that the improved decoding algorithm mentioned in this paper improves the time-consuming decoding algorithm of the original algorithm matrix, and it also improves the efficiency. Thus, the improved algorithm is efficient and simple.

For the two algorithms of merger decoding and improved decoding algorithm, we compare their efficiency by computing time and use EVENODD code to carry out experiments. Through the cyclic iteration, the merging decoding and the improved decoding algorithm, the experimental conditions of the data loss are hypothesized and the lost data are reconstructed gradually. Finally draw the experimental results as shown in Figure 2.

It can be seen clearly from the above figure that the

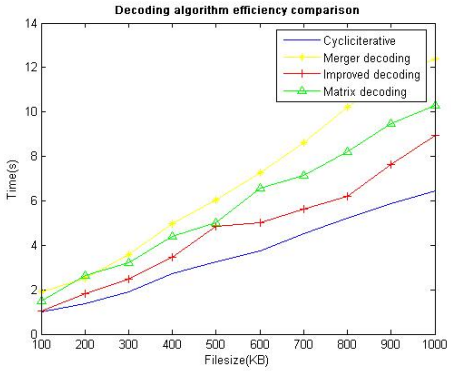


Figure 2: Four algorithm efficiency comparison

cyclic iterative decoding algorithm has the highest efficiency. The improved decoding algorithm takes longer than the cyclic iterative decoding algorithm, but the difference is not much. And because the merger decoding is used in the process of inversion calculation, it will be twice as much as the cyclic iterative decoding algorithm. It can be seen that the efficiency of the improved algorithm proposed in this paper is not bad.

Next, take RDP-code as an example to conduct double fault experiments. Four different algorithms are used respectively. Finally, the experimental diagram is shown as follows in Figure 3:

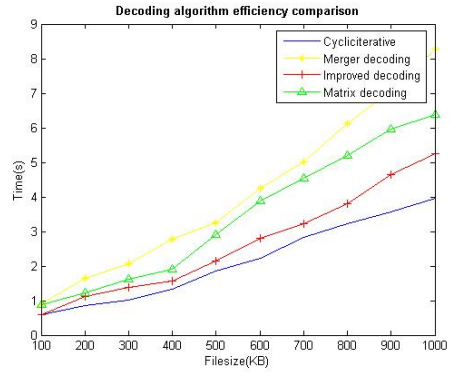


Figure 3: Four algorithm efficiency comparison

It can be seen that the above results are similar to those of EVENODD-code, so the algorithm proposed in this paper is more efficient.

5 Summary

In this paper, an improved erasure code decoding algorithm based on matrix decoding algorithm is proposed to recover the random sector loss of erasure codes. It is applicable to any theory recoverable situation, which includes the situation that the original algorithm can not be recovered, and it continues the good generality of the original algorithm. It is found that the efficiency of the algorithm is more efficient than the original one, and the calculation efficiency is higher, which can be widely used in the

random sector loss. This improved algorithm proposed in this paper is currently running on the binary matrix array code for the operation. After that, this algorithm can be extended to non-binary decoding operations, such as RS code.

Acknowledgments

The author thanked Pro.Dan Tang for the affirmation and guidance of the algorithm, the research of this paper can not be separated from Professor Tang's continuous encouragement and correction.This work was supported by the National Natural Science Foundation of China under Grant No. 61501064 and Science and technology program of SICHUAN 2018GZ0099.The authors gratefully acknowledge the anonymous reviewers for their valuable comments.

References

[1] M. Blaum, J. Brady, J. Bruck, and Jai Menon, "Evenodd: An efficient scheme for tolerating double disk failures in raid architectures," *IEEE Transactions on Computers*, vol. 22, vol. 2, pp. 245–254, 1994.

[2] P. Corbett, B. English, A. Goel, T. Grcanac, S. Kleiman, J. Leong, and S. Sankar, "Row-diagonal parity for double disk failure correction," in *Usenix Conference on File and Storage Technologies*, pp. 1, 2004.

[3] J. L. Hafner, V. Deenadhayalan, K. K. Rao, and J. A. Tomlin, "Matrix methods for lost data reconstruction in erasure codes," in *Conference on Usenix Conference on File and Storage Technologies*, pp. 14, 2005.

[4] J. L. Hafner, V. Deenadhayalan, T. Kanungo, and K. K. Rao, "Performance metrics for erasure codes in storage systems," *Chaos An Interdisciplinary Journal of Nonlinear Science*, vol. 18, vol. 3, pp. 150–156, 2004.

[5] C. Huang and L. Xu, "Star: An efficient coding scheme for correcting triple storage node failures," *IEEE Transactions on Computers*, vol. 57, vol. 7, pp. 889–901, 2008.

[6] M. S. Hwang, T. H. Sun, C. C. Lee, "Achieving dynamic data guarantee and data confidentiality of public auditing in cloud storage service," *Journal of Circuits, Systems, and Computers*, vol. 26, no. 5, 2017.

[7] E. Mugisha, G. Zhang, M. Zine, E. Abidine, and M. Eugene, "A TPM-based secure multi-cloud storage architecture grounded on erasure codes," *International Journal of Information Security and Privacy*, vol. 11, no. 1, pp. 52-64, 2017.

[8] J. Qureshi and A. Malik, "On optimization of wireless xor erasure codes," *Physical Communication*, vol. 27, no. 1, pp. 74–85, 2018.

[9] D. Tang, "Research of methods for lost data reconstruction in erasure codes over binary fields," *Journal of Electronics and Technology*, vol. 14, no. 1, pp. 43–48, 2010.

[10] P. Teng, L. Chen, D. Yuan, and X. Wang, "Sparse random erasure code: A large-scale data storage disaster recovery method," *Journal of Xi'an Jiaotong University*, vol. 51, no. 5, pp.48-53, 2017.

[11] P. Teng, J. Zhang, L. Chen, and X. Wang, "Random array code: A highly disaster-tolerant and expandable raid storage disaster recovery method," *Engineering Science and Technology*, vol. 49, pp. 3, pp. 110–116, 2017.

[12] Y. Wang, F. Xu, and X. Pei, "Research on erasure-code fault-tolerance technology in distributed storage," *Journal of Computer*, vol. 1, pp. 236–255, 2017.

[13] 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.

[14] S. Yang and G. Zhang, "A survey of data repair methods in erasure code storage system," *Computer Science and Exploration*, vol. 11, no. 10, pp. 1531–1544, 2017.

Biography

Di Fan, master candidate. She is currently an Master student in Chengdu University of Information Technology, Chengdu, China. Her research interests include coding theory and information security.

Feng Xiao, Master candidate. He is currently an Master student in Chengdu University of Information Technology, Chengdu, China. His research interests include coding theory and database theory.

Dan Tang received his Ph.D. degree from Graduate University of Chinese Academy of Sciences (CAS), Beijing, China in 2010. He is currently an associate professor with Chengdu University of Information Technology, Chengdu, China. His research interests include coding theory and secret sharing scheme.

Computer Real-Time Location Forensics Method for Network Intrusion Crimes

Yingsu Qi

(Corresponding author: Yingsu Qi)

Beijing Police College, Beijing, China

No. 11, Nanjian Road, Nankou Town, Changping district, Beijing 102202, China

(yingsq_ys@yeah.net)

(Received May 31, 2018; Revised and Accepted Oct. 13, 2018; First Online Mar. 6, 2019)

Abstract

In recent years, with the development of Internet technology, people's life has been more convenient with the help of the Internet. But it has also given rise to a new form of crime, Internet crimes. Computer forensics is thus born to deal with new kinds of crimes. In this study, the improved K-means clustering algorithm was adopted to obtain computer real-time evidence of network intrusion crimes. The detection performance of the improved algorithm for the four types of characteristic data was analyzed by MATLAB. Moreover, the detection performance of traditional clustering algorithm under different intrusion attack modes was compared. The results demonstrated that the improved algorithm is more suitable for detecting the first kind of characteristic data; and compared with the traditional clustering algorithm, under the three flood attack modes of User Datagram Protocol (UDP), Internet Control Messages Protocol (ICMP) and Transmission Control Protocol (TCP), the improved algorithm is better and has faster speed of data processing. In conclusion, the improved K-means clustering algorithm can be applied to the computer real-time location and forensics of network intrusion crimes.

Keywords: Clustering Algorithm; Computer Forensics; Network Intrusion; Network Security

1 Introduction

With the globalization of information, the Internet has developed rapidly. Enterprises, governments and individuals cooperate with each other on the Internet, which improves work efficiency as well as convenience and enriches people's spare time life [7]. The Internet has the advantages of openness and sharing. On the one hand, with these two advantages, the network has a variety of resources and facilitates people's life. On the other hand, since these two advantages are not only for the general public, cybercriminals can also take advantage of these two advantages to launch an invasion against enterprises, governments or individuals through the Internet.

In recent years, the popularity of mobile terminals that can be connected to the Internet has also diversified the ways of network crimes. Due to the convenience brought by the network, it brings greater security risks to users [6]. Therefore, in order to protect the network security of users and combat the illegal criminal acts with the help of the network, computer forensics technology emerges at the historic moment. Computer forensics technology [11] intercepts and analyzes network intrusion data to obtain relevant criminal evidence. The early computer forensics technology is usually used to reverse and analyze the intrusion scene after the event to obtain relevant evidence. However, with the rapid development of network technology, the corresponding traces of the perpetrators are often eliminated after the invasion, making it difficult to obtain effective evidence for the early static forensics [10]. Therefore, the real-time dynamic forensics technology that can collect effective evidence in large-scale networks is required.

Here are the relevant studies. Sun *et al.* [12] proposed a modeling method for Evolved Packet Core (EPC) network intrusion system in order to obtain appropriate defense strategies. Firstly, the attack behavior in the network is described, and the intrusion attack is introduced into the model based on time. Finally, UPPAAL is used to verify the correctness of the proposed model. Compared with other typical attack models, the proposed model is comprehensive and integrated. Malialis *et al.* [8] proposed a method combining task decomposition, team rewards and punishments and forms of rewards and punishments, which is called differential rewards and punishments system. The system learns quickly. The extensibility of the algorithm is demonstrated in experiments involving 1000 learning agents. Compared with baseline technology and throttling technology, this method is more effective. It can adapt to complex attack rates and dynamics, and it is robust to agent failure.

Hodo proposed the use of Artificial Neural Network (ANN) to deal with malicious attacks in the Internet [3].

The classification of normal mode and threat mode in network is studied. The experimental results show that this method has 99.4% accuracy and can detect all kinds of DDoS attacks successfully. In this paper, the improved k-means clustering algorithm is adopted to obtain computer real-time evidence of network intrusion crimes. The detection performance of the improved algorithm for the four types of characteristic data was analyzed by simulation on MATLAB. At the same time, the detection performance of traditional clustering algorithms under different intrusion attack modes was compared.

2 Computer Forensics Technology

With the rapid development of network technology, people's life style has changed dramatically. The emergence of the Internet not only provides convenience, but also provides a new criminal channel which is different from the traditional one. In view of the new network crime, the computer forensics technology arises at the historic moment. Computer forensics is also called network forensics. The evidence of network crime provided by it is recognized by law and can be used as an effective and reliable basis in judgment. The key principle of computer forensics [13] is to use the relevant network data analysis technology to judge the motive of the crime and to collect the network crime data of the suspect specifically, which is taken as the evidence of legal trial.

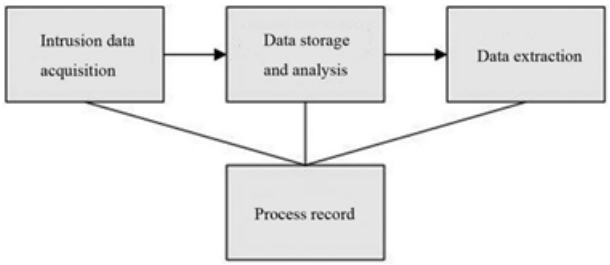


Figure 1: Technical process of computer forensics

As shown in Figure 1, the process of computer forensics technology is divided into three steps [4]. First, data for an intrusion is acquired. Then the collected data is preserved and analyzed. Finally, the evidence and data obtained from the analysis are extracted for a summary report. In the three processes of evidence collection, based on the continuity principle of effective evidence, it is necessary to record the three processes.

In terms of network intrusion data, network intrusion is usually an attack on a server with a large amount of garbage data, and the actual purpose of the operation is hidden in it for an invasion purpose. The common network attack mode is a flood attack that is simply rough and difficult to defend. The common flood attack modes are User Datagram Protocol (UDP) flood attack, Transmission Control Protocol (TCP) flood attack and Internet Control Messages Protocol (ICMP) flood attack. The above flood attacks are all DDoS flood attacks, which flood the server with a large amount of garbage data, causing the server to run out of resources and go down. The difference between the three DDoS flooding attacks mentioned above lies in the different message data used for the attack. UDP message mainly attacks target IP address. TCP mainly uses the TCP protocol to suspend a service. Information for requesting confirmation is continuously sent to consume server resource. And access to other users is blocked . ICMP is a traffic attack that consumes resources by sending more than 65,535 bytes of data to the server.

In order to ensure the effectiveness of computer forensics, the following three principles should be followed [2]:

1) The crime scene is made of a timely blockade to ensure the preservation of evidence maximization;

2) The collection of evidence network integrity and continuity must be guaranteed. The final evidence submitted shall be the same as the final evidence investigated;

3) In the whole process of evidence collection, there must be a third party to supervise so to ensure the impartiality of evidence.

3 Cluster Detection Algorithm Based on Network Packet Analysis

When the intrusion data is analyzed, the corresponding intrusion detection algorithm should be applied. In this paper, clustering detection algorithm based on network packet is selected. The working principle of clustering algorithms is simply to classify the collected data and distinguish normal data from abnormal data. In practical work, the collected data are first converted into vectors, and then the vector data are classified by clustering algorithms. There are different clustering algorithms for different data types. In this paper, k-means clustering algorithm [1] is selected and its principle is as follows:

$$\begin{aligned} d(a,b) &= d_{continuous}(a,b) + d_{discrete}(a,b) \\ a &= \{a_1, a_2, \cdots, a_i, a_{i+1}, a_{i+2}, \cdots, a_n\} \\ b &= \{b_1, b_2, \cdots, b_i, b_{i+1}, b_{i+2}, \cdots, b_n\} \end{aligned} \quad (1)$$

where $d(a,b)$ is the distance between eigenvectors a and b; $d_{continuous}(a,b)$ is the distance of continuous eigenvectors; $d_{discrete}(a,b)$ is the distance of discrete eigenvectors, and $a_1, a_2, \cdots, a_i$ and $b_{i+1}, b_{i+2}, \cdots, b_n$ are continuous eigenvectors; $a_i, a_{i+1}, a_{i+2}, \cdots, a_n$ and $b_1, b_2, \cdots, b_i$ are

discrete eigenvectors. Moreover,

$$\begin{aligned} d_{continuous}(a,b) &= \sqrt{\sum_{j=1}^{i}(a_j - b_j)^2} \\ d_{discrete}(a,b) &= \sum_{j=i+1}^{n} d(a_j, b_j) \\ d(a_i, b_i) &= 0 \text{ if } a_i = b_i \\ d(a_i, b_i) &= 1 \text{ if } a_i \neq b_i. \end{aligned} \quad (2)$$

Then according to the calculated distance between the two eigenvectors of a and b, the similarity of the two features is determined, and the classification is carried out accordingly.

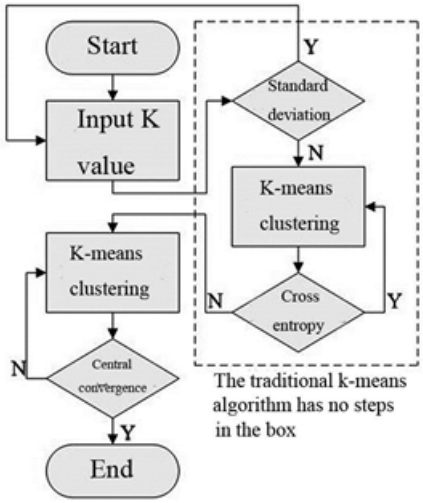


Figure 2: The process of the improved k-means algorithm

As shown in Figure 2, K represents the number of categories. In the traditional k-means algorithm process, there are no steps in the dotted box. The clustering effect of data depends on whether the choice of K value is reasonable. When the detected data is discrete, the defect exposure is more obvious. In this paper, the data of computer forensics for network intrusion is discrete. Therefore, in order to improve the accuracy and efficiency of forensics, combined with the theory of hierarchical clustering, the traditional algorithm is improved by increasing standard deviation and cross entropy [9]. The formula of its increase steps are:

Standard deviation is:

$$\sigma = \sqrt{\frac{d(a_i, C)}{m}} \quad (3)$$

where C is the threshold; M is the number of categories.

Cross entropy is:

$$D_R = -\log(\frac{1}{mn}\sum_{i=1}^{n}\sum_{j=1}^{m} e^{\frac{-||a-b||^2}{2\sigma^2}}), \quad (4)$$

where n is the number of another category.

The improved K-means algorithm is shown in Figure 2. First, K value and eigenvector data are input to calculate its center point. Then K is divided into the right number of species by the standard deviation. Next, it is aggregated by cross entropy. After completion, the classification is carried out through the k-means algorithm until the convergence of K value is stable, and the algorithm is completed.

4 Simulation Test

4.1 Data Preparation

KDD99 data set was used [14]. Data in the data set can be divided into four categories according to characteristics. The first category is a description of the characteristics of the server status. The second category is a description of the characteristics of the operations received by the server. The third category is a description of the characteristics of network traffic in the past 2 s. The fourth category is a description of the characteristics of network traffic over the first 100 connections. There are totally 40 features. Each data in the data set has 42 dimensions. The first 41 dimension is the characteristic attribute of the data, and the last one is the decision attribute, which indicates whether the data is abnormal and is used to detect the performance of the algorithm. The data set including data and normal data of the known network intrusion type is used to simulate the real network environment. In this paper, 100,000 pieces of data were selected in the data set, of which 30,000 pieces were normal data, and the rest were network intrusion data.

4.2 Data Preprocessing

If the weight of eigenvalues in the data is too large, it will affect the accuracy of the detection algorithm. Therefore, it is necessary to normalize the characteristic data. The formula [5] is:

$$x' = \frac{x - N_{mn}}{N_{max} - N_{min}}, \quad (5)$$

where x is the characteristic data that needs to be normalized; N_{min} is the minimum value in the characteristic data; and N_{max} is the maximum value in the characteristic data. When the maximum and minimum are equal, the normalized eigenvalue is denoted as 0.

4.3 Experimental Environment

In this study, the algorithm model is written by Matlab simulation platform, and the experiment is carried out on the laboratory server. The server is configured as: Windows 7 system, I7 processor, 16G memory.

4.4 Experimental Settings

1) The improved k-means algorithm is used to detect the data set where the initial K value is set as 6; the initial standard deviation is set as 0.6; the splitting parameter c_1 is set as 2.1, and the aggregation parameter c_2 is set as 0.4.

2) Packet uploads in the network are simulated as attacks. The traditional k-means algorithm and the improved k-means algorithm are used to detect the intrusion data, and each attack mode is simulated 4,000 times.

4.5 Evaluation Criteria

The performance evaluation of intrusion detection algorithm is usually represented by three data which are accuracy rate B_C, false alarm rate E_A and failure rate d. For the performance of intrusion detection algorithm, the higher the accuracy, the better; and the lower the false alarm rate and failure rate, the better. The calculation formula is as follows:

$$B_C = \frac{C_P + C_N}{C_P + C_N + M_P + M_N}, \quad (6)$$

$$E_A = \frac{M_N}{C_N + M_N}, \quad (7)$$

$$N_A = \frac{M_P}{C_P + M_P}, \quad (8)$$

where C_P is the attack data correctly classified; C_N is normal data with correct classification; M_N is the normal data with misclassification, and M_P is a misclassified attack data.

4.6 Experimental Results

4.7 Detection Performance of the Improved K-means Algorithm

As shown in Figure 3, the detection accuracy rate of the improved k-means algorithm for the intrusion of Data I is 97%. The false alarm rate is 11.1%, and the missing alarm rate is 5.6%. The detection accuracy rate of the intrusion of Data II is 94%. The false alarm rate is 12.3%, and the missing alarm rate is 6.2%. The detection accuracy rate of the intrusion of Data III is 93.1%. The false alarm rate is 13.5%, and the missing alarm rate is 7.1%. The detection accuracy rate of the intrusion of Data IV is 89%. The false alarm rate is 13.9%, and the missing alarm rate is 8.2%. It can be intuitively seen from the Figure 3 where the improved K-means algorithm has the highest detection accuracy and the lowest false alarm rate and omission rate on Data I. Compared with the other three types of data, the improved K-means algorithm is more suitable for detecting the network intrusion attacks of Data I.

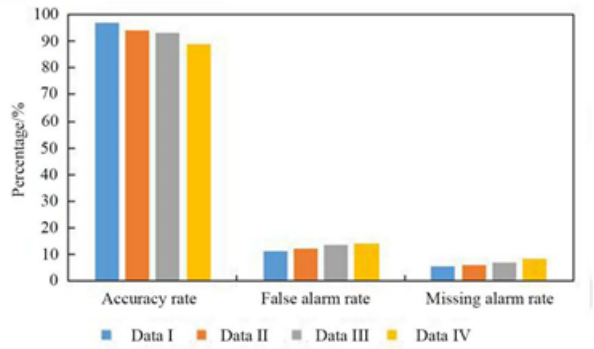


Figure 3: Intrusion detection results of the improved K-means algorithm

4.8 Performance Comparison with Traditional K-means Algorithm

As shown in Figure 4, in the UDP simulated flood attack mode of Data I, the detection accuracy rate of the traditional K-means algorithm is 91.4%. The false alarm rate is 9.8%, and the missing alarm rate is 7.8%. The detection accuracy rate of the improved k-means algorithm is 93.2%. The false alarm rate is 4.5%, and the missing alarm rate is 3.1%.

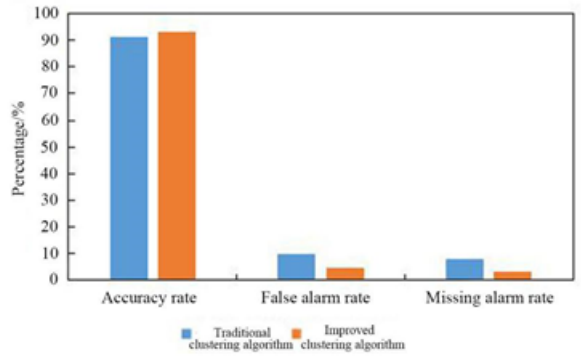


Figure 4: Detection performance of the two algorithms in UDP flood attack mode

As shown in Figure 5, in the ICMP simulated flood attack mode of Data I, the detection accuracy rate of the traditional K-means algorithm is 91.3%. The false alarm rate is 9.6%, and the missing alarm rate is 7.5%. The detection accuracy rate of the improved k-means algorithm is 93.5%. False alarm rate is 4.6%, and missing rate is 2.9%. It can be seen that the improved clustering algorithm has better detection performance for ICMP flood attack mode.

As shown in Figure 6, in the TCP simulated flood attack mode of Data I, the detection accuracy rate of the traditional K-means algorithm is 91.2%. The false alarm rate is 9.9%, and the missing alarm rate is 7.9%. The detection accuracy rate of the improved k-means algorithm

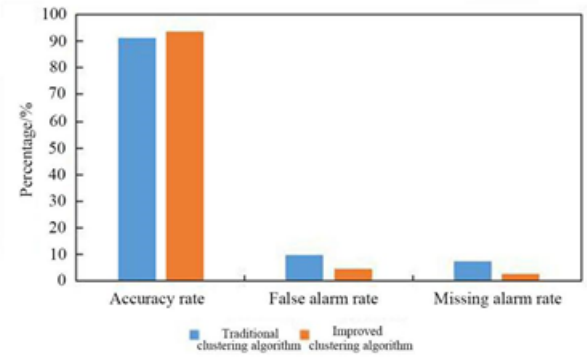


Figure 5: Detection performance of the two algorithms in ICMP flood attack mode

is 94.1%. The false alarm rate is 4.8%, and the missing alarm rate is 3.3%. It can be seen that the improved clustering algorithm has better detection performance for TCP flood attack mode.

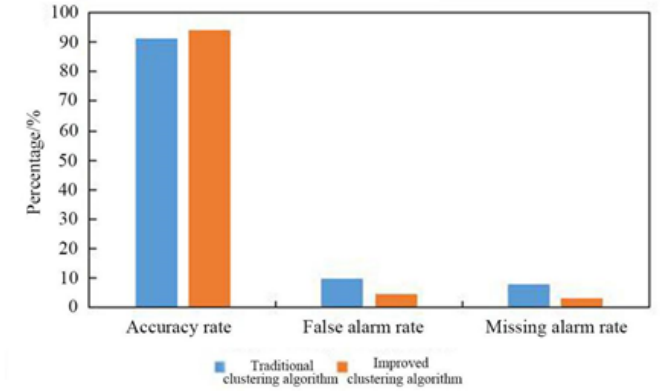


Figure 6: Detection performance of the two algorithms in TCP flood attack mode

As shown in Table 1, the processing speed of traditional clustering algorithm for UDP flood attack is 331.5 piece/s. The processing speed of ICMP flood attack is 321.6 piece/s. The processing speed of TCP flood attack is 335.4 piece/s. The processing speed of the improved clustering algorithm for UDP flood attack is 523.6 piece/s. The processing speed of ICMP flood attack is 528.4 piece/s. The processing speed of TCP flood attack is 531.3 piece/s. It can be seen from the statistical results that the improved clustering algorithm is significantly faster than the traditional clustering algorithm in processing the intrusion data. Moreover, according to the comparison of the amount and time of the evidence of the intrusion data obtained by computer forensics in the past network intrusion crimes, it can be seen that the processing performance of the improved clustering algorithm for the intrusion data exceeds the actual performance requirements, and the algorithm has been able to be used for the real-time forensics of network intrusion crimes.

5 Conclusion

In this study, the improved K-means clustering algorithm was adopted to obtain computer real-time evidence of network intrusion crimes. In addition, the detection performance of the improved algorithm for the four types of characteristic data was simulated and analyzed in MATLAB. Moreover, the detection performance of traditional clustering algorithm under different intrusion attack modes was compared. The performance of the improved k-means clustering algorithm in detecting the first type of feature data is better than the other three. Under the flood attack modes of UDP, ICMP and TCP, the improved k-means clustering algorithm is superior to the traditional clustering algorithm in the detection performance of intrusion data. The two algorithms were compared in the processing speed of the intrusion data under the three intrusion attack modes, and the improved algorithm obviously exceeds the traditional clustering algorithm.

References

[1] L. Duan, F. Yu, L. Zhan, "Improved fuzzy c-means clustering algorithm," in *International Conference on Natural Computation, fuzzy Systems and Knowledge Discovery*, IEEE, pp. 44–46, 2016.

[2] D. Gugelmann, F. Gasser, B. Ager, *et al.*, "Hviz: HTTP(S) traffic aggregation and visualization for network forensics," *Digital Investigation*, vol. 12, pp. s1-s11, 2015.

[3] E. Hodo, X. Bellekens, A. Hamilton, *et al.*, "Threat analysis of IoT networks using artificial neural network intrusion detection system," in *International Symposium on networks*, IEEE, pp. 6865–6867, 2016.

[4] F. Karpisek, I. Baggili, F. Breitinger, "WhatsApp network forensics: Decrypting and understanding the WhatsApp call signaling messages," *Digital Investigation*, vol. 15, pp. 110-118, 2015.

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

[6] S. Khan, A. Gani, A. W. Wahab, *et al.*, "Network forensics: Review, taxonomy, and open challenges," *Journal of Network & Computer Applications*, vol. 66, pp. 214–235, 2016.

[7] M. Korczynski, A. Hamieh, J. H. Huh, *et al.*, "Hive oversight for network intrusion early warning using DIAMoND: A bee-inspired method for fully distributed cyber defense," *IEEE Communications Magazine*, vol. 54, no. 46, pp. 60–67, 2016.

[8] K. Malialis, S. Devlin, D. Kudenko, "Distributed reinforcement learning for adaptive and robust network intrusion response," *Connection Science*, vol. 27, no. 3, pp. 19, 2015.

Table 1: Processing speed of the two algorithms in different attack modes

Detection algorithm	Intrusion attack mode	Number of invasions	Total detection time/s	Processing speed (piece/s)
Traditional clustering algorithm	UDP flood attack	5216	15.73	331.5
	ICMP flood attack	4215	13.11	321.6
	TCP flood attack	4856	14.48	335.4
Improved clustering algorithm	UDP flood attack	5745	10.97	523.6
	ICMP flood attack	4351	8.23	528.4
	TCP flood attack	4951	9.32	531.3

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

[10] A. Singhal, C. Liu, D. Wijesekara, "POSTER: A logic based network forensics model for evidence analysis," in *ACM Sigsac Conference on Computer and Communications Security*, pp. 1677–1677, 2015.

[11] G. Singhchhabra, P. Singh, "Distributed network forensics framework: A systematic review," *International Journal of Computer Applications*, vol. 119, no. 19, pp. 31–35, 2015.

[12] Y. Sun, T. Y. Wu, X. Q. Ma, H. C. Chao, "Modeling and verifying the EPC network intrusion system - based on timed automata," *Journal of Pervasive and Mobile Computing*, vol. 24, pp. S1574119215001145, 2015.

[13] M. Vallentin, R. Sommer, R. Sommer, "VAST: A unified platform for interactive network forensics," in *Usenix Conference on Networked Systems Design and Implementation*, pp. 345–362, 2016.

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

Biography

Yingsu Qi, female, born in July 1979, is a master of engineering and lecturer. Her research interests include computer forensics and information security. She has presided over the project of Beijing Education Committee's Talent Program and published articles including Computer Forensics Technology, Thoughts on Network Quality Training from the Perspective of Personal Information Security and Research on Enterprise Information Sharing Platform Based on Real-time Database and XML and participated in the compilation of a ministerial textbook Handbook of Public Security Information Application Law.

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.