IJNS
International Journal of Network Security

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

Vol. 17, No. 6 (Dec. 2015)

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. 17, No. 6 (Dec. 1, 2015)

An Efficient Key Management Scheme in Multi-Tier and Multi-Cluster Wireless Sensor Networks

Manivannan Doraipandian[1], P. Neelamegam[2]
(Corresponding author: Manivannan Doraipandian)
School of Computing, SASTRA University[1]
Tirumalaisamudram, Thanjavur, Tamil Nadu 613401, India
School of Electrical and Electronics Engineering, SASTRA University[2]
(Email: dmv@cse.sastra.edu)
(Received Apr. 16, 2013; revised and accepted Aug. 15 & Nov. 26, 2013)

Abstract

Wireless Sensor Network is a collection of autonomous sensor nodes placed spatially. Unlike wired networks the sensor nodes here are subject to resource constraints such as memory, power and computation constraints. Key management and Security are the area of research in WSN. To ensure high level security encryption is necessary. The strength of any encryption algorithm depends upon the key used. So Key Management plays a significant role. The proposed KMS using LLT matrix achieves both Node-to-Node communication and Group communication. The main objective of the scheme is to strengthen the data transferring security mechanisms and also to ensure efficient key generation and management along with authentication. The main feature of this proposed system is 100% Local-connectivity; efficient node revocation methodology, perfect resilience; three-level authentication cum key generation and the most importantly reduced the storage. The scheme and its detailed performance analysis are discussed in this paper.

Keywords: Cholesky decomposition, key connectivity, resilience, WSN

1 Introduction

WSN [12] is a collection of nodes from hundreds to thousands. Each node has processing units, sensing unit and power source usually the battery. Sensor nodes are resource constrained in terms of computation, memory. Because of its transmission nature and also because of its deployment in hostile environments, security mechanisms available for wired ad-hoc networks are not applicable for WSN. So new security mechanisms [9] should be introduced but satisfy the security requirements such as authentication, confidentiality, integrity and availability. Though many cryptographic algorithms are available, but the strength of the algorithm purely depends on the key used. For eg. If AES is incorporated, whoever involved in building up the security mechanisms knows about the AES. So the importance will be on key and also the size of the key. If 128 bit key is used, a possible set of key will be in 2128. So to establish a secure communication key management plays a vital role. Key management includes key generation, distribution and storage of keys. The attackers usually made an attack on the key management level rather than cryptographic algorithm level. Since the sensor node is resource constrained designing a key management scheme for WSN is challenging issue. In recent years, many key management schemes are proposed. Key management schemes are broadly classified into three categories: key pre-distribution, arbitrated key mechanisms and self-enforcing mechanisms. Arbitrated keying mechanisms depend upon trusted third party agent. Of that if the node gets compromised all information about the network will get revealed. Self-enforcing mechanism is a public key cryptography method. Since sensor nodes are resource constraining this method is not preferable.

Almost all key management schemes [1, 2, 3, 6, 7, 10, 16] are based on key pre-distribution method in which keys are loaded into sensor nodes before deployment. Designing a suitable key management scheme for all kinds of WSN organization such as hierarchical or distributed is another challenging issue. Based upon applications and architecture used KMS has to be defined. Once after designing the KMS, the metrics [12] to be evaluated against KMS is security (Authentication, resilience, node revocation), efficiency (memory, processing, bandwidth, energy, key connectivity) and flexibility (deployment knowledge, scalability). Satisfying all the metrics in a single key management scheme is difficult. If suppose group key communication is incorporated, periodic updating of group key is necessary. This increases expenses on rekeying. Thus

in this situation key connectivity is not an issue rather rekeying is. The evaluation metrics are mainly based on the architecture and keying mechanism used.

2 Related Works

Some schemes follows partial pairwise key methodology [5]. For a network with n nodes, it is not necessary to store n-1 keys in each node to achieve a connected graph. The degree of each node is determined by the probability of connectivity. Usually it is expected to be high. Basic Scheme [9], Q-composite [3] are based on this method. These schemes undergo three steps: Key Pre-Distribution phase; shared-key phase and path key establishment phase. Bloms [1] scheme; Du-et-al [6] multiple space, LU [12] schemes are also pair-wise schemes. Instead of storing the keys directly [18], corresponding rows and columns of the matrix are stored [14] and pairwise key is generated using vector multiplication whenever two nodes want to communicate. Most of the hierarchical network schemes [8] use group key mechanisms. Resilience and node revocation becomes an issue. Many schemes [17] are introduced without deployment knowledge. As a result, the probability to nodes to be within each others communication is less. Thus the connectivity is less. Considering all these factors, the PROPOSED scheme is a mixture of pairwise, group, matrix-based, hierarchical network with deployment knowledge.

3 Our Contribution

The proposed scheme is a matrix based scheme. A symmetric matrix is decomposed into two matrices using CHOLESKY factorization. It is almost similar to LU scheme. The reason for choosing CHOLESKY factorization is that: the two matrices are lower triangular matrix and its transpose. This reduces STORAGE, COMPUTATION and COMMUNICATION overhead to a large extent. It is enough to store only the row values unlike LU Scheme where in row and corresponding columns are stored. This reduces the MEMORY CONSUMPTION to half of that consumed in LU [12, 16] scheme.

The proposed scheme uses the HIERARACHICAL NETWORK STRUCTURE to enhance DIVISION OF LABOUR. Processing and work decrease down the group. This gives a clear idea of what type of nodes to be used at which level. Two types of keys used in the proposed scheme: pairwise key [3, 5, 10] and the other Group key. Group key is mainly used for commenting purposes. Pairwise keys serve two purposes. Firstly, they are used in message passing then its for node revocation. The corresponding group head will initiate the node revocation. This involves deleting records pertaining to the captured node and replacing the old group row with new one. This cannot be multi-cast because even the captured node will receive the message. Thus the pairwise key between a node and the group head is used for it.

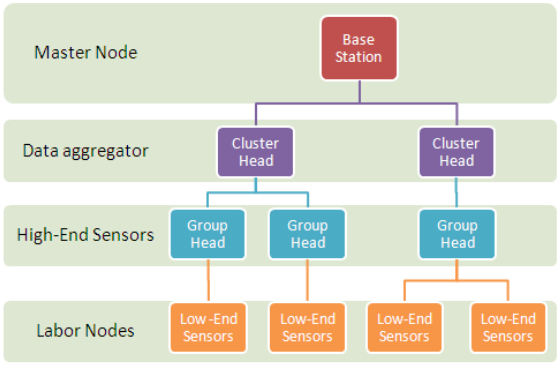

Figure 1: Layered clustered architecture - WSN

Proposed scheme also uses message passing efficiently by providing three level authentications cum key generation mechanism within three steps.

4 The Proposed Scheme

4.1 Architecture

The four layered clustered architecture comprises of Base station at the top level, Cluster heads at the second level, High end sensors as the group head in the third level and Low end sensor nodes at the bottom level (see Figure 1).

The main advantage of a multi-layered clustered architecture [11] is that the number of keys loaded in each sensor nodes will be appreciably less compared to distributed sensor networks. The hierarchical architecture enhances the scalability of the system. Further it provides Division of Labor system where in each node is loaded with optimal work it can perform [7]. Thus, hierarchical clustered architecture [13] gives a clear idea of what type of nodes to be used at different levels. The main objective in WSN is to achieve 100% connectivity at low power consumption. To achieve maximum communication range the node consumes maximum transmission power and thus the range of communication is traded-off with energy consumption. Typically, WSN nodes are expected to work efficiently at low power consumption. Hence decreasing the power consumption cuts down the communication range.

The nodes communicate with other nodes without any nodes intervention without regarding the transmission power of communication [15]. The main advantage of this is the security which is 100%, further; the data received is a primary data. Also there is minimal possibility of data loss. But still this is not welcomed in WSN because of its high energy consumption. Here for 100% connectivity all the nodes should be in the communication range of other nodes, this limits the network coverage.

These multi-hop techniques are used. The communication range of a node is reduced subject to the energy consumption constraint. In order to achieve 100% connectivity it is not necessary for all nodes to be within the

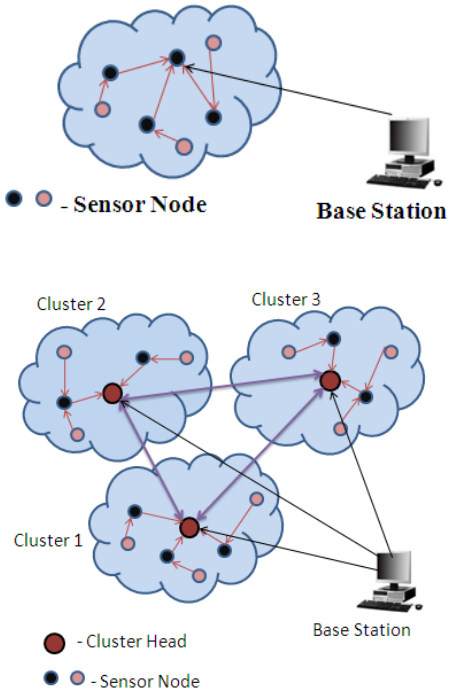

Figure 2: Direct and multi hop communication - WSN

communication range of nodes (see Figure 2).

It can be achieved using multi-hopping where in the two nodes, which are out of their communication range, communicated via the nodes present within the range. The identification of such nodes leads to path establishment phase. The main disadvantage of this is data loss and insecurity. Since the communication range [17] of a node is reduced keeping in mind the power consumption, multi-hop techniques are used for long range communication. Here low end sensor nodes are grouped under High end Group heads (nodes). If one node in a group wants to communicate with the node in the other group multi-hopping is done via their respective group heads. By doing this data is more secure as the receiving node knows the source of the message, also always there exist exactly two nodes (the respective group heads) in between the sender and the receiver. Thus the proposed scheme uses the optimized connectivity with minimal power consumption as the criterion for grouping nodes. The same criterion is followed for clustering group heads under powerful cluster heads. The cluster heads communicates directly with the base station. The proposed multi-layered clustered architecture is hence the best architecture.

4.2 Communication Flow

Base Station: Full duplex communication between base station and cluster heads.

Cluster Head: Full duplex communication between cluster heads (inter), base station and group heads (intra).

Group Head: Full duplex communication between group heads (inter), cluster head and sensor nodes (intra).

Sensor Nodes: Full duplex communication between nodes and group heads (within a group) [13, 14, 15, 18].

4.3 Outline of the Scheme

All the sensor nodes are loaded with the programs and data before deployment. Based on the locality of deployment, the sensor nodes are grouped under High end sensor nodes. Further the groups are clustered (depending upon the structure of deployment) under cluster head. Thus the knowledge about the locality of sensor nodes is known in advance. The base station is fed with the details of all sensor nodes, group as well as cluster heads such as Number and IDs of all nodes belonging to a group; Number and IDs of all group head belonging to a cluster; Number of clusters and ID of each cluster head. Lower Triangular matrices decomposed from a symmetric matrices form the basis of key generation. The symmetric decomposition is done using CHOLESKY decomposition [15].

Assume there are c clusters, g groups, and n nodes. Thus $c \times c$ symmetric matrix is allotted for inter-cluster communication along with the base; $c(g \times g)$ symmetric matrices for inter grout (intra cluster) communication within a cluster; $g(n \times n)$ symmetric matrices for inter node (intra group) communication. The trick of the trade is that the values of the order of the symmetric matrices are kept as large as possible. This is done to achieve better scalability. Using separate sets of matrices for different layers of architecture, different sets of keys are generated for each layer. Each layer is a completed graph with m nodes (m is appreciably less than the order of the symmetric matrix allotted to it). For commanding purpose say from base station to cluster heads or from cluster heads to its group heads or from group heads to its nodes a unique key is generated at each level. A key array consisting of possible keys with which a node can communicate is stored in its memory. This ensures authenticated communication between nodes. A Common hashing array for generating indices is used for encrypting the message. Periodically checks are made by the respective heads to test whether a node is alive or dead.

4.4 System Components and Functionalities

Base Station. This is the master node of the network. It is at the topmost level of the architecture [18]. It commands and controls all its co-ordinate nodes. It receives the aggregated data from various cluster heads and processes it [7, 18]. It stores cluster IDs, group IDs, number of clusters, number of groups in a cluster, number of nodes in a group along with their

IDs. Further it stores one row of the $c \times c$ matrix for establishing pairwise key between cluster heads for inter cluster communication; one common row from the $c \times c$ matrix for broadcasting. It also stores Key pool, hashing array [11].

Cluster Head. This node serves two purposes: one is that it reduces the burden of the base station by performing data aggregation and distribution of messages from/to various group heads; the other is that it aids inter-group communication by acting as a mediator [7, 18]. Moreover it initiates group head revocation. It stores Cluster IDs, Group IDs, and number of groups under its control. One row of $c \times c$ matrix for inter cluster communication; one row from the allotted GXG matrix for intra- cluster (inter-group) communication; the common row stored in base station (to receive message broadcasted by the base station) and one common row of $g \times g$ matrix for broadcasting (to group heads) purpose. Further it also stores the hashing array and key pool list.

High End Sensor Nodes/Group Heads. This node plays the role of cluster heads at this level, i.e., it performs data aggregation and distribution of messages from/to its nodes. This also takes the role of initiating node revocation [7, 18]. It stores IDs of node belonging to it, IDs of group heads belonging to same cluster. One row of GXG matrix for inter group communications; one row of NXN matrix for intra-group (inter-node) communication; the common row of GXG matrix stored in its cluster head (to receive the message broadcasted by cluster head); one common row of NXN matrix for broadcasting (to nodes) purpose. Further it also stores the hashing array and key pool list [4, 13].

Low End Sensor Nodes (simple called nodes). This is the working node of the system, which senses and transmits sensed data to its group heads. Group ID, one row of the $n \times n$ matrix for communicating with group head; the common row stored in its group head (to receive the message broadcasted by it). Further it also holds the hashing array and a key list with two elements one the key value for communicating to group head and the other for receiving the broadcasted message [4, 13].

NOTE: Different sets of matrices are dedicated to different cluster heads. Though the hashing array stored in the nodes is same for all, the key pool list varies in accordance with the matrices allotted to it.

4.5 Key Management

4.5.1 Symmetric Matrix Decomposition

The methodologies used in this scheme are listed as follows:

Cholesky factorization. LU decomposition constructs both lower and upper triangular factors **L** and **U** Cholesky decomposition constructs a lower triangular matrix **L** whose transpose $\mathbf{L}^T$ itself an upper triangular matrix such that A=LL^T.

Cholesky Factorization Algorithm. If the order of the A matrix is N then,

1) Set $k = 1$;
2) Repeat the following until $k <= N$;
3) For K^{th} $N \times N$ Matrix:
   a. $a_{k,k} = \sqrt{a_{k,k}}$;
   b. $a_{k+1:N,k} = a_{k+1:N,k}/a_{k,k}$;
   c. $a_{k+1:N,K+1} = a_{k+1:N,K+1} - a_{k+1:N}, \ k * a_{k+1,k}$;
   d. $a_{k+2:N,k+2} = a_{k+2:N,k+2} - a_{k+2:N,k} * a_{k+2,K}$ and so on;
   e. Increment k by 1.

4.5.2 Pre-deployment Phase

All the parameters that are mentioned in the system and component phase are loaded to the appropriate nodes.

4.5.3 Key-establishment Phase

After successful deployment of nodes establishing connectivity is the crucial step. This is done using keys. In simple words two nodes can communicate if and only if they share a common key.

4.5.4 Pair-wise Key Establishment

Steps involved in pair-wise key establishment [10] between two nodes:

1) The sender node A sends its row R_{na} in format I node to the receiver node B.
   Message Format I: NodeID$_B$ || row_values || hashing_index (base) || hashing_index (shift) || NodeID$_A$.

2) Node B receiver the messages and retrieves the row values of A. It computes the Key K_{AB} and checks it presence in the Key pool. If it is present then Node B sends its row, checked bet, hash of the key in format 2 to A.
   Message Format II: NodeID$_A$ || row_values || hash(key) || checked_bht || hashing_index(base) || hashing_index(shift) || NodeID$_B$.

3) Node A receives the message and retrieves row values R_{nb} of B, key value K_{AB} and computes the key value, K_{BA} using R_{na} and R_{nb}. Then it checks whether K_{BA} is present in its key pool and also r_{eA} matches with K_{AB}. If it matches node A sends the message to B using shh computed key.

4.5.5 Group Key Establishment

Group key are mainly used for commanding and controlling the nodes. There is only one group key at any level. Here broadcasting technique is followed.

Steps involved in group key establishment:

1) The group head broadcasts the message to all its nodes using message format I mentioned below.
Message Format I: Group ID || row_values || hash on key || cipher message || hashing_index (base) || hashing_index (shift).

2) Appropriate nodes receive the message and retrieve the necessary command.

The process of decryption is as same as the process depicted above in pair wise key establishment.

4.5.6 Cluster Key Establishment

This process is similar to that of group key establishment.

5 Performance Analysis

The following are the factors (affecting performance of the system) that are analyzed in this phase. These are (1) Key connectivity; (2) Efficiency (Computation, Storage, Communication); (3) Scalability.

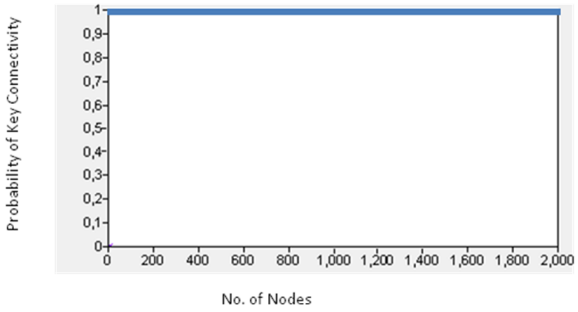

Figure 3: Key connectivity vs. Number of nodes

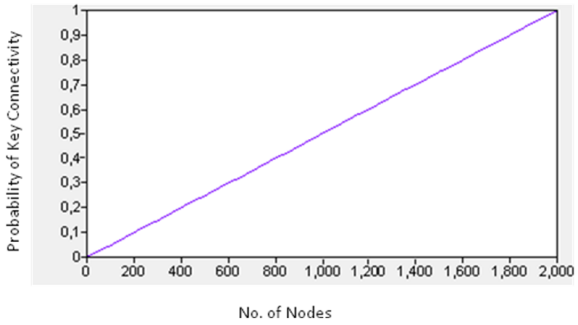

Figure 4: Key connectivity vs Number of hops; RP $n = 2000$, p vs m

5.1 Key Connectivity

It is a measure of the possibility of communication between two nodes in a network; this is usually referred to as local connectivity [1, 3, 6]. Global connectivity is a measure of connected components in the entire network. For system with high performance key connectivity should be high. This is because with high connectivity probability of multi-hopping reduces. This reduces unnecessary intermediate communications which in turn reduce the transmission power. Thus battery power (power source of sensor nodes) is reserved for processing and hence performance increases with key connectivity.

In the proposed scheme, 100% Key connectivity is achieved at each tier of the hierarchy, i.e., the network is a completely connected graph at each level of hierarchy (completely pairwise) as shown in Figure 3. The proposed network (structure) is a connected (not a fully connected) graph. As mentioned the connected components of the graph are fully connected. Generally, a lot of communication happens only within nodes of the same level, i.e., the number of intra-level communications is more when compared to inter-level communication. Thus it is enough if 100% key connectivity is assured within a level and inter-level communication can be achieved using secondary or ternary neighbors. The proposed scheme uses this strategy.

Random pairwise scheme: In order to reduce the Key storage when compared to EG [9] scheme, the entire graph is divided into several overlapping connected components (nodes). Though this scheme does not support 100% connectivity but ensures the network is connected. Since all the nodes perform the same tasks, frequent communication between them is required. Thus for two nodes, which are far apart, to communicate lots of hopping has to be done, this increases communication overhead. Thus high key connectivity is achieved at the expense of transmission Power.

For RP scheme the key Connectivity will be $p = (1/n) * m$ where p denotes a probability of connectivity; n denotes number of nodes; m denotes a degree of each node. The key connectivity for RP scheme is shown in Figure 4.

Asymmetric Pre-distribution scheme: The key connectivity is not 100% initially. Whenever two non-connected nodes want to communicate, they first establish a pair-wise key between them with their first degree H sensor node. Thus Key connectivity gradually reaches 100% at the expense of memory, i.e., the storage memory in L sensor nodes inner-cases.

The Key connectivity for AP scheme is $1-((p-m)!(p-l)!/p!(p-m-l)!))$ where p = pool size; e = number of keys in H - Sensor node; l = number of keys in L- Sensor node. In the proposed scheme 100% key connectivity is achieved between primary neighbors, unlike AP scheme as shown in Figure 5. Thus a balance is stroked between Key connectivity and Key storage.

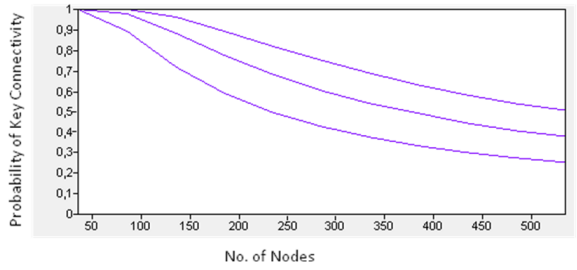

Figure 5: key Connectivity for AP scheme

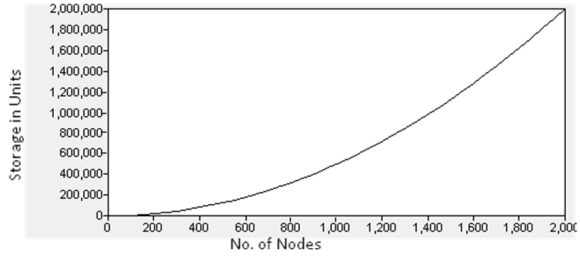

Figure 6: Storage in proposed scheme

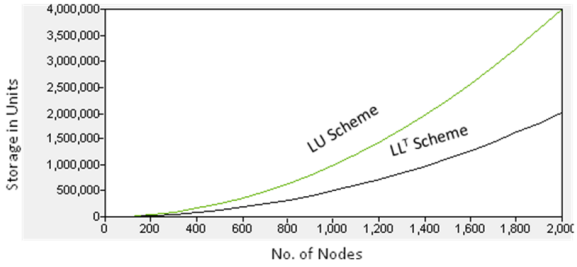

Figure 7: Storage in LU vs. Proposed Scheme

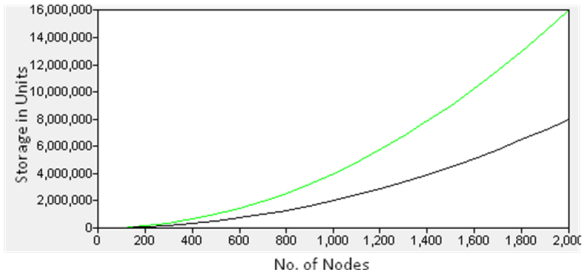

Figure 8: Number of nodes vs Memory

5.2 Efficiency

5.2.1 Storage

A typical node is subject to memory constraints for better performance, i.e., a maximum storage capacity of a node is generally small. Also high end nodes have better storage capacity compared to low end sensor nodes. The proposed scheme uses this fact and stores data accordingly, i.e., the storage decreases down the hierarchy. Matrix generation and other major storing activities are limited with the top level itself.

Proposed scheme: The Storage will be $(x/2)(x+1)$ where x is the number of Nodes and the respective graph is shown in Figure 6. In LU [10] Decomposition each and every node stores one row of Lower Triangular matrix and corresponding column of upper matrix to generate keys by matrix multiplication. In the proposed scheme the upper triangular Is the transpose of the lower triangular matrix ($U = L^T$). This reduces the number of rows to be stored in each node to one. Let minimum size of one row be on an average 4 bytes and say there are 5000 nodes (as in a typical network); LU utilizes 40000 (2*4*5000) bytes whereas the proposed scheme consumes only half the above value, i.e., 20000 bytes (4*5000). The remaining reserved memory is efficiently for authentication and computational purposes.

The storage for LU will be x*(x+1) whereas in LL^T it will be $(x/2)(x+1)$ where x be the number of nodes (see Figure 7).

In Random Pairwise [10] scheme the voting keys, that are stored in low end sensor nodes, used for node revocation increases the storage in the nodes. In the proposed scheme any node revocation within a group or a cluster is initiated and taken care by their corresponding group heads of cluster heads, imposing no additional memory consumption. Head nodes being a high end sensor node can store additional information. Thus the network objectives are achieved subject to memory constraints without any degradation in performance.

In Du et al. scheme τ distinct keys spaces from the possible choices (say w) are randomly loaded into the nodes. The size of one row is λ+1, thus for each node (λ +1) τ units are required.In the Proposed scheme many entries in lower triangular matrix are zero thus size of one row is far less than that used in Du et al. scheme. This strategy helps to reduce memory consumption to a large extent.

In LEAP each and every node is loaded with individual key, pairwise key, group key and cluster key to achieve high connectivity between different levels of hierarchy. The proposed scheme uses only pairwise key and group key to achieve the connectivity that LEAP achieves. This reduces the memory consumption to almost half of that in LEAP (see Figure 8).

5.2.2 Computation

Computation is done at the expense of power consumption. Since a node is expected to work with minimal power consumption too much computation degrades nodes performance. The proposed scheme basically involves three computations multiplication, one-level base conversion,

shifting. Multiplication is done for key generation. The base for conversion is chosen in such a way that it terminates at one level itself, thus restricting the number of divisions to one. Shifting being a bit twiddling operation does not consume much power. Thus the computational power consumption is relatively less compared to many schemes and also the proposed scheme does not perform any computation for node authentication. Also since most of the row entries are zero computation becomes simple.

In Polynomial based scheme the nodes are supposed to compute their key using n-degree polynomial functions with two variables which involve computing exponential powers of those variables and their summation. This consumes a lot power. Proposed scheme limits the number of arithmetic computations to one or two and mainly performs simple bit twiddling operations and hence consumes relatively less power. In Blom [10] and Du et al. scheme the nodes compute keys by multiplying rows and columns. To reduce storage on each node, only the seed of the column (Vander monde matrix with seed s) is stored. But this imposes computational overhead in generating the column which happens at the expense of power consumption. The proposed scheme has no such overhead in generating column as only rows are stored.

5.2.3 Communication

Communication is directly related to transmission power. Thus for high performance unnecessary communications should be avoided. In the proposed scheme the nodes are loaded with all the possible keys with which it can communicate, keeping in mind the transmission power, before deployment. Many schemes have shared-Key discovery phase and path-key establishment phase. This involves a lot of communication between nodes. The proposed scheme being completely pre-deployed does not involve any communication of this type. Thus saving a lot of transmission power. Further the proposed scheme involves only two communications for key generation and node authentication.

In the AP scheme [7], Du et al. matrix scheme, Blundo, Liu and Ning scheme, q-composite scheme involves both shared key establishment And path key establishment phase which increase communication. In LU scheme the key computation involves three steps whereas proposed scheme uses only two steps, hence 33% transmission power is saved. Also in LU if there is a key mismatch then the authentication mechanism (μ TESLA) is initiated which consumes computation power. The proposed scheme does node authentication and key establishment within the two steps and hence is efficient.

5.3 Scalability

This measures the performance of the network in addition of new nodes. For a typical network the performance should not be affected while adding new nodes. In the proposed scheme the order of the matrix is set to the maximum having a futuristic view on the scalability. When a node is added to a particular group, unused row of the allotted matrix is loaded into it along with all other remaining necessary data before deployment. The strategy used here for accommodating a large number of new nodes is to group them under new group. This strategy handles scalability to a large extent.

In Hierarchical LU the rows and columns are randomly loaded into the nodes. When more and more new nodes are added the possibility of two nodes having same rows increases. Thus probability of link compromising increases. In RP scheme each node is loaded with m identifiers in its vicinity. When a new node is added, its key identity must be updated in that connected component of the network. This imposes communication overhead.

6 Security Analysis

6.1 Resilience

A network should be secure enough so that the entire message passing is done secretly. This ensures no data leakage. Thus a malicious user cannot hack the information from the nodes in the network. Resilience is a measure of how quickly the system recovers upon node capturing. The recovery depends on the impact of node capturing on the system, i.e., it indirectly measures how much remaining nodes and links get compromised on node capturing. A good wireless sensor network must definitely be resilient, otherwise, the entire system will be attacked and all the data can be hacked out of it.

6.2 Message Interception

This is a situation where malicious users intercept messages (brute force attacks) in the network by snoops, traffic analysis, modification, masquerading, repudiation, replaying, and denial of service and many other security threats and attacks. A good network is supposed to ensure high data integrity, confidentiality and availability.

The proposed scheme produces cipher messages which are highly encrypted. Thus any malicious user will not be able to retrieve any information from it. By doing this the proposed scheme overcomes the traffic analysis and spoofing threats. Here the node ID and the computed key values which are checked against a key pool list acts as the digital signature to provide authentication. Further, the proposed scheme uses encipherment and thus ensures data integrity and overcomes masquerading, modification, etc.

Assuming the awkward situation where in the attacker retrieved the content of the message and found the key. In the proposed scheme each and every node has a unique pairwise key with each node within a level. The attacker remains helpless with one key as he will not be able to masquerade with other nodes. Thus no link gets compromised. The only link that gets compromised is the link from which the attacker retrieved the information.

In such a situation the corresponding group or cluster head initiates node recovery mechanism after identifying malicious network interfaces. Heads replace the row values and key pool list of all the nodes in the vicinity of the alien interface with new rows from its allotted matrix and a corresponding key pool list. This is done by unicasting all this information to respective nodes.

6.3 Node Compromising

Here the attacker uses any physical attacks to directly capture nodes. On capturing a node the attacker will get to know about all the information that is stored in it.

Considering an awkward situation where in a node in the proposed network is captured physically and all the information that is stored in it are known to the attacker. Using this node the attacker can easily communicate with all the other nodes in its communication range. If the node is a group head then the attacker will be able to retrieve information from the entire cluster in which it belongs. In the proposed scheme the respective group head or cluster head immediately initiates recovery mechanism after identifying the attacked node, giving no room for the attacker to extract information from other nodes. Once the attacked node is known by the group head it initiates node revocation mechanism. Its first and foremost task is to transmit the node ID of the attacked node to all the remaining nodes in the cluster. This is done by unicasting to the nearest node, the attacked node ID, the new row for group communication and change in hashing index in hashing format (mentioned above). The nearest node retrieves all the necessary information and performs three basic operations. Firstly it passes this message to its neighbor node which is not the attacked one. Secondly, it deletes the attacked node ID and its corresponding key value from the list it stores. It then changes the row for group communication and updates the old key value for group communication in the key list with the new computed group key. Thirdly, the node changes the range of base and shifting number generated by the pseudo random generator. Thus though the attacker who knows the message format can't hack it because he doesn't know to what number the index is referring to. All the other nodes also do the same. Further this ensures resilient property in the network.

In the AP scheme, Hierarchical LU scheme, Du et al. [17] matrix scheme the probability of using the same key to establish links between different nodes are more because the rows are randomly loaded into the nodes, i.e., two or more nodes may have same row values for shared key establishment. Thus when one link gets compromised it will also affect all the other links which used the same row for generations. In RP scheme the voting keys play a crucial role in node revocation. This increases storage in every node. Thus resilience is achieved at the expense of storage. Also voting leads to communication overhead.

In LU scheme [16] the probability of two nodes to have same row value increases with an increase in the number of nodes and number of keys chosen for the scheme. Thus number of link compromising increases with increase in the number of keys in a node as shown in Figure 9. strategy helps to reduce memory consumption to a large extent.

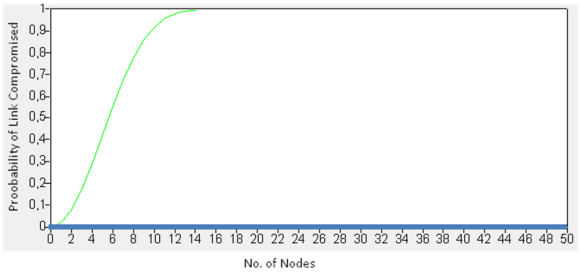

Figure 9: Number of nodes vs probability of link compromised

6.4 Node Authentication

Authentication [13, 14] ensures that both the parties at the ends of the communication are authenticated. In the proposed scheme, no additional authentication mechanism, like μ-Tesla, is used. Instead a part of the key generation mechanism is used for authenticating. First the receiver node that extracts the ID from the message checks with its node ID. If it matches then it is confirmed that the message is from authenticated node. Further confirmation is done by checking the computed key value with the key pool list it possesses. If there is a match then authenticated communication takes place between them. In cases of Mismatch in either of the steps, the corresponding group or cluster head is alerted by the node in which mismatch occurred. The heads probe into this issue and finds whether mismatch is due to loss of data or due to malpractices. Thus two level authentications cum key generation reduces communication to a large extent.

In many schemes such as q-composite scheme, SHELL, the keys are directly deployed in the nodes. The nodes there directly send messages using those keys. In order to authenticate nodes some additional mechanisms are needed. But only a few schemes incorporate such mechanisms. Thus attacking such networks with less or authentication is simple. Authentication ensures data integrity and confidentiality in a network.

7 Summary and Conclusion

An Efficient Key Management scheme for WSN with multi-tier and multi clustered architecture using LLT is discussed LLT matrix will play a vital role to achieve FULL local key connectivity with less communication and less computation overhead. This Proposed protocol is an efficient, secured, scalable and multilevel authenticated between nodes, nodes to group head, group head to clus-

ter head and cluster head to cluster head. In this architecture, Choleskey decomposition constructs a lower triangular matrix L, whose transpose LT itself an upper triangular matrix such that A=rL^T. Using this technique pairwise key establishment phase, Group key establishment phase, cluster key establishment phase is achieved. Performance analyzes in terms of key connectivity, efficiency, scalability are done and the results are noted. The security analysis are made related to resilience and node authentication and the results are noted finally the comparison is made between proposed scheme with an existing key management scheme in terms of performance and security analysis. The summary of the results is discussed in Table 1. The results indicate that the proposed scheme is well suited for dynamic homogeneous and heterogeneous sensor networks.

Table 1: Summary and result

Metrics	Achievements
Key Connectivity	100% Local key Connectivity
Storage	Less Storage;
Communication	No shared key and path key establishment phase
Computation	Bit twiddling operation reduces computation overhead
Scalability	Unused rows from the matrix is loaded
Resilience	Changing the range of base and shift number
Authentication	Multi-Tier authentication, where node ID's acts as a digital signature

References

[1] R. Blom, "An optimal class of symmetric key generation systems", in *Advances in Cryptology (EUROCRYPT'84)*, LNCS 209, pp. 335–338, 1985.

[2] C. Blundo, A. De Santis, A. Herzberg, S. Kutten, U. Vaccaro, M. Yungcet, "Perfectly secure key distribution for dynamic conferences", *Information and Computation*, vol. 146, no. 1, pp. 1–23, 1998.

[3] H. Chan, A. Perrig, D. Song, "Random key predistribution schemes for sensor networks", in *Proceedings of the 2003 IEEE Symposium on Security and Privacy*, pp. 197–213, 2003.

[4] C. C. Chang, L. Harn, and T. F. Cheng, "Notes on polynomial-based key management for secure intra-group and inter-group communication", *International Journal of Network Security*, vol. 16, no. 2, pp. 143–148, Mar. 2014.

[5] H. Dai, H. Xu, "Key pre-distribution approach in wireless sensor networks using LU matrix", *IEEE Sensors Journal*, vol. 10, no. 8, pp. 1399–1409, Aug. 2010.

[6] W. Du, J. Deng, Y. S. Han, P. K. Varshney, J. Katz, A. Khalili, "A pairwise key predistribution scheme for wireless sensor network", *ACM Transactions on Information and System Security*, vol. 8, no. 2. pp. 228–258, 2005.

[7] X. Du, Y. Xiao, M. Guizani, and H. H. Chen, "An effective key management scheme for heterogeneous sensor networks", *Ad Hoc Networks*, vol. 5, no. 1, pp. 24–34, 2007.

[8] M. Eltoweissy, H. Heydari, L. Morales, H. Sadborough, "Combinatorial optimization of key management in group communications", *Journal of Network and Systems Management*, vol. 12, no. 1, pp. 33–50, 2004.

[9] L. Eschenauer, V. D. Gligor, "A key management scheme for distributed sensor networks", in *Proceedings of the 9th ACM Conference on Computer and Communication Security (CCS'02)*, pp. 41–47, 2002.

[10] A. A. Kamal, "Cryptanalysis of a polynomial-based key management scheme for secure group communication", *International Journal of Network Security*, vol. 15, no. 1, pp. 68–70, 2013.

[11] D. Macedonio, M. Merro, "A semantic analysis of key management protocols for wireless sensor networks", in *Science of Computer Programming*, pp. 53–78, Elsevier Science, 2014.

[12] D. Manivannan, R. Ezhilarasie, P. Neelamegam, K. R. Anuj, "An efficient and hybrid key management scheme for three tier wireless sensor networks using LU matrix", in *Proceedings of the First International Conference on Advances in Computing and Communications (ACC'11)*, pp. 111–121, Kochi, India, 2011.

[13] O. K. Sahingoz, "Large scale wireless sensor networks with multi-level dynamic key management scheme", *Journal of System Architecture*, vol. 59, pp. 801–807, 2013.

[14] Qi Shi, N. Zhang, M. Merabti and K. Kifayat, "Resource-efficient authentic key establishment in heterogeneous wireless sensor networks", *Journal of Parallel and Distributed Computing*, vol. 73, no. 2, pp. 235–249, 2013.

[15] M. Turkanovic, B. Brumen and M. Hölbl, "A novel user authentication and key agreement scheme for heterogeneous ad hoc wireless sensor networks, based on the internet of things notion", in *Ad Hoc Networks*, pp. 96–112, Elsevier Science, 2014.

[16] Mi Wen, Y. Zheng, H. Li, K. Chen, "A hierarchical composition of LU matrix-based key distribution scheme for sensor networks", in . *Emerging Technologies in Knowledge Discovery and Data Mining*, LNCS 4819, pp. 608–620, Springer, 2007.

[17] M. F. Younis, K. Ghumman, M. Eltoweissy, "Location-aware combinatorial key management scheme for clustered sensor networks", *IEEE Transactions on Parallel and Distributed Systems*, vol. 17, no. 8, pp. 865–882, 2006.

[18] S. Zhu, S. Setia, and S. Jajodia, "LEAP: Efficient security mechanisms for large-scale distributed sensor networks", *ACM Transactions on Sensor Networks*, vol. 2, no. 4, pp. 500–528, 2006.

Manivannan Doraipandian obtained Ph.D. degree from SASTRA University, Thanjavur in 2013. Since 1996, he has been in the teaching profession and currently he is a Senior Assistant Professor in the Department of Computer Science, School of Computing, SASTRA University, Thanjavur, Tamil Nadu, India. His area of interest include Cryptography, Security in Embedded Systems, Wireless Sensor Networks using ARM processors and Embedded Communication Systems.

P. Neelamegam obtained Ph.D. degree from Bharathidasan University, Tiruchirappalli in 1992. Since 1971, he has been in the teaching profession and currently he is a Professor in the Department of Electronics and Instrumentation, School of Electrical and Electronics engineering, SASTRA University, Thanjavur, Tamil Nadu, India. His research interests include Microprocessors, Microcontrollers, embedded system based Instrumentation, wireless sensor network, neural network and fuzzy logic.

Blind Expressive Ciphertext Policy Attribute Based Encryption for Fine Grained Access Control on the Encrypted Data

Xingbing Fu[1], Shengke Zeng[2], and Fagen Li[1]
(Corresponding author: Xingbing Fu)
School of Computer Science and Engineering, University of Electronic Science and Technology of China[1]
No.2006, Xiyuan Avenue, West Hi-tech Zone, Chengdu, 611731, P.R.China
(Email: fuxbuestc@126.com)
School of Mathematics and Computer Engineering, Xihua University[2]
Chengdu, Sichuan, China
(Received February 1, 2015; revised and accepted May 1 & May 15, 2015)

Abstract

Oblivious transfer with access control is a protocol where data in the database server are protected with access control policies and users with credentials satisfying the access policies are allowed to access them, whereas the database server learns nothing about the data accessed by users or about her credentials.Our scheme has the advantages as follows: First, our scheme maintains the privacy property of oblivious transfer and offers access control mechanism. Second, it allows the expressive access control polices that directly supports **AND**, **OR** and **Threshold** gates. Third, the communication complexity in our scheme is constant in the numbers of records which have been accessed. Fourth, our scheme is constructed in prime order bilinear group.

Keywords: Access control, bilinear maps, ciphertext policy attribute based encryption, linear secret sharing, oblivious transfer, standard model

1 Introduction

With the advent of cloud computing, more and more organizations plan to adopt the cloud computing service. However, the concerns for the security and privacy make them hesitate to adopt this service. While the encryption techniques can be employed to protect the outsourced data, the cloud service providers can still collect the sensitive information on who accesses the outsourced data, and how she accesses them. To address the problem, researchers proposed to employ the oblivious transfer (OT, for short) [7] primitive to preserve the users privacy. However, oblivious transfer in its basic form has no access control functionality, that is, the users can obtain any files chosen by them without any restrictions. To distinguish the authorized users from the unauthorized users, access control mechanisms are introduced in such a way that only the authorized users are allowed to access data, whereas the unauthorized users cannot. However, traditional access control mechanisms assume the items being requested are knowledgeable.

To preserve the users' privacy and let access control mechanism be enforced by the service provider (database), researchers proposed oblivious transfer with access control mechanism which, for each record of the database, defines an access control policy that determines the attributes, roles and rights which the user needs to possess to access this record. To meet the requirements for the maximal amount of privacy, this mechanism should provide guarantees as follows:

1) The record can be accessed by only the authorized users.
2) Which record the user has accessed is not learned by the database provider.
3) The database provider does not learn which attributes the user possesses when the database is accessed by her.
4) Access control mechanism should be flexible enough to enforce different expressive access control policies.

An encryption scheme is employed to securely share data among users. The symmetric cryptography and traditional public key cryptography are suitable for the setting in which a user securely share data with another user that is known to her in advance, that is, the communication model is one-to-one.Furthermore,access to the encrypted data is all or nothing-a user is either able to decrypt and obtain the entire plaintext or she does not learn anything at all about the plaintext except for its length.

With the advent of cloud computing, where there exist a large number of users, the traditional cryptosystem is insufficient. For instance, the data provider may want to share data according to some policy based on the recipient's credentials or attributes and only the data users satisfying the policy can decrypt. The traditional public key cryptosystem cannot handle such tasks.

Sahai and Waters [17] first proposed the Attribute Based Encryption (ABE) scheme to handle the aforementioned problem. In their scheme, the private keys and ciphertexts are associated with attribute sets, and a private key can decrypt a ciphertext iff there exists a match between the attributes of the private key and those of the ciphertext. Decryption is enabled only if at least d attributes overlap between a ciphertext and a private key. Their scheme is useful for error-tolerant encryption with biometrics, while their scheme is limited to handling threshold access structure. Since the Attribute Based Encryption scheme is proposed, different ABE schemes and their applications [15, 12, 6] are presented in terms of flexibility, efficiency, and security. Existing ABE schemes are classified as Key Policy ABE (KP-ABE) schemes [10, 16] and Ciphertext Policy ABE (CP-ABE) schemes [2]. In KP-ABE schemes, keys are associated with access policies, and ciphertexts are identified with attribute sets.Iff the keys associated with access policies satisfied by the attributes associated with the ciphertexts are able to decrypt the ciphertexts. In CP-ABE schemes, access policies are associated with the ciphertexts and keys are associated with attributes. If and only if keys associated with attributes satisfying the access policy associated with the ciphertext are able to decrypt it.

In the `CP-ABE` schemes, the data are protected with access polices, and only those users whose attributes satisfy the access policies are able to decrypt to access them.BSW scheme [2] are the first to implement CP-ABE scheme which is expressive and efficient attribute based encryption scheme. However,security proof of their scheme are based on the generic group model which assumes that an adversary needs to access an oracle to perform any group operations. To achieve ciphertext policy attribute based encryption scheme in the standard model, work has been done as follows: Cheung and Newport [5] proposed a CP-ABE scheme which construct a policy with an **AND** gate under the bilinear Diffie-Hellman assumptions.However, their scheme requires that the number of system attributes be fixed at setup and the access structure of their scheme only support an **AND** gate. These two drawbacks make it less expressive. To enhance the expressiveness, Goyal, Jain, Pandey, and Sahai [9] proposed Bounded CP-ABE scheme in the standard model. However, the encryption and decryption complexity blows up by an $n^{3.42}$ factor in the worst case, which limits its usefulness in practice.Lewko et al. [13] proposed a CP-ABE scheme in the standard model which is expressive, and adaptively secure. However, their scheme is based on composite order bilinear group which incurs some efficiency loss and assumption is non-standard assumption.To overcome this problem, Waters [19] present a CP-ABE scheme which is both expressive and is proven secure under a standard assumption in the standard model.Our scheme builds on this scheme.

We propose blind expressive ciphertext policy attribute based encryption scheme to achieve fine grained access control over the encrypted data. Our scheme has the advantages as follows: First, our scheme maintains the privacy property of oblivious transfer and offers access control mechanism.Second,it allows the expressive access control polices that directly supports **AND**, **OR** and **Threshold** gates.Third, the communication complexity in our scheme is constant in the numbers of records which have been accessed.Fourth, our scheme is constructed in prime order bilinear group.

The remainders of our paper are organized as follows: We discuss related work in Section 2. We introduce preliminaries in Section 3. We present scheme definition, security game and Blind CP-ABE scheme in Section 4. We present the scheme construction in Section 5. Blind Private Key Generation Protocol is presented Section 6. We propose fully simulatable oblivious transfer with fine grained access control in Section 7. The performance of our scheme is evaluated in Section 8. We conclude and specify the future work in Section 9.

2 Related Work

Coully et al. [7] presented a scheme based on state graphs where users obtain credentials binding them to a particular state in the graph. This scheme limits the possible transitions between states to enforce access control. Their scheme has the advantages as follows: (1) It can be applied to different oblivious transfer schemes; (2) It permits the access control policies to be changed without changing the database. Unfortunately, their scheme has the two following drawbacks: (1) Each time users access the database, they have to obtain a new credential. (2) This scheme cannot efficiently express a large class of access policies based on state graphs. Camenisch et al. [3] presented an oblivious transfer with access control mechanism in which each user can obtain a credential certifying whether she possesses some attributes used to describe each record of data. A user can access the record as long as she possesses these attributes, which makes access policies only support `AND` condition. To support access policy in disjunctive form, database server needs to duplicate the record, which increases the size of database. To directly support access policy in disjunctive form, Zhang et al. [20] present oblivious transfer with access control which realizes disjunction without duplication.Their scheme builds on fully secure attribute based encryption scheme proposed by Lewko et al. [13]. However, their scheme is based on composite order bilinear group which results in some efficiency loss. Furthermore, their scheme does not perform key sanity check and ciphertext sanity check. In case the issuer and the database are malicious, the two users

which possess the same attributes may decrypt the same encrypted record to different plaintext record, which does not guarantee anonymity of the users.

3 Preliminaries

3.1 Bilinear Map

Let $\mathbb{G}$ and $\mathbb{G}_T$ denote two cyclic groups whose order is prime order p, and g, u are a generator of $\mathbb{G}$, respectively. e is a bilinear map $e : \mathbb{G} \times \mathbb{G} \to \mathbb{G}_T$ that has the properties as follows:

Bilinearity. for any $a, b \in \mathbb{Z}_p$, $e(g^a, u^b) = e(g, u)^{ab}$.

Nondegenerate. $e(g, g) \neq 1_{\mathbb{G}_T}$, $e(g, g)$ is a generator of $\mathbb{G}_T$. If the group operation on $\mathbb{G}$ and on the bilinear map $e : \mathbb{G} \times \mathbb{G} \to \mathbb{G}_T$ are efficiently computable, then $\mathbb{G}$ is a bilinear group. Our scheme employs the symmetric bilinear map such that: $e(g^a, u^b) = e(g, u)^{ab} = e(g^b, u^a)$.

3.2 Access Structure

Let $\mathbb{S}$ be the universe of attributes. An access structure [1] on $\mathbb{S}$ is a collection $\mathbb{A}$ of non-empty subsets of attributes, i.e., $\mathbb{A} \subseteq 2^{\mathbb{S}} \setminus \{\}$. We call the sets in $\mathbb{A}$ the authorized attribute sets, and the sets not in $\mathbb{A}$ the unauthorized attribute sets. Specifically, an access structure is monotone if $\forall B, C$: if $B \in \mathbb{A}$ and $B \subseteq C$, then $C \in \mathbb{A}$. In this scheme, only monotone access structure is handled.

3.3 Linear Secret Sharing Scheme

A secret sharing scheme [1, 4, 18] Π over the attribute set is called linear over $\mathbb{Z}_p$ if (1) The shares for each attribute of a secret form a vector over $\mathbb{Z}_p$. (2) There is a matrix M with h rows and n columns for Π. For any $j = 1, \cdots, h$, let the function φ defined the attribute that labels the j^{th} row as $\varphi(j)$. Given the column vector $\overrightarrow{v} = (s, x_2, \cdots, x_n)^T$, in which T is the transpose of the vector $\overrightarrow{v}$, s is the secret that will be shared, and $x_2, \cdots, x_n \in \mathbb{Z}_p$ are uniformly at random picked, then $M\overrightarrow{v}$ is the vector of h shares of the secret s based on Π. The share $(M\overrightarrow{v})_j$ belongs to the attribute $\varphi(j)$.

Let attribute set $S \in \mathbb{A} \wedge S \in \mathbb{S}$ be any authorized attribute set, and let $J = \{j | j \in \{1, \cdots, h\} \wedge \varphi(j) \in S\}$. Then, there exist constants $\{\eta_j \in \mathbb{Z}_p\}_{j \in J}$ such that, if $\{s_j\}_{j \in J}$ are valid shares of a secret s according to Π, then $\Pi_{j \in J} \eta_j s_j = s$.

3.4 Commitment Scheme

A commitment scheme comprises the three algorithms as follows:

Setup $(1^\kappa) \to \mathbb{CP}$**.** This algorithm takes in a security parameter κ, and it outputs the commitment parameters $\mathbb{CP}$.

Commit $(\mathbb{CP}, m) \to (\mathcal{C}, \mathcal{D})$**.** This algorithm takes in the commitment parameters $\mathbb{CP}$ and a message m, and it outputs a pair $(\mathcal{C}, \mathcal{D})$.

Decommit $(\mathbb{CP}, m, \mathcal{C}, \mathcal{D}) \to \{0, 1\}$**.** This algorithm takes in $\mathbb{CP}$, m, $\mathcal{C}$, $\mathcal{D}$, and it outputs 1 if $\mathcal{D}$ opens $\mathcal{C}$ to m, else 0.

We employ the Pedersen commitment scheme [14], where the commitment parameters are a group whose order is prime order p, and random generators $(h_0, \cdots, h_\lambda)$, where λ is a positive integer. In order to commit to the values $(a_1, \cdots, a_\lambda) \in \mathbb{Z}_p^\lambda$, select a random $\varpi \in \mathbb{Z}_p$ and set $\mathcal{C} = h_0^\varpi \prod_{i=1}^{\lambda} h_i^{a_i}$ and $\mathcal{D} = \varpi$.

3.5 Zero Knowledge Proof

We employ definitions from [8]. A pair of algorithms (P, V) which interact with each other is a proof of knowledge (POK) for a relation $R = \{(\gamma, \delta)\} \subseteq \{0,1\}^* \times \{0,1\}^*$, where knowledge error is $\lambda \in [0, 1]$ if (1) For all $(\gamma, \delta) \in R$, $\mathtt{V}(\gamma)$ accepts a conversation with $\mathtt{P}(\delta)$ with probability 1; (2) There is an expected PPT algorithm KE, called the *knowledge extractor*, such that if a cheating prover $\widehat{P}$ has probability ε of convincing V to accept γ, the KE, when given rewindable black box access to $\widehat{P}$, outputs a witness δ for γ with probability $\varepsilon - \lambda$.

A proof system (P, V) is perfect zero-knowledge if there is a PPT algorithm Sim, the *simulator*, such that for any PPT cheating verifier $\widehat{V}$ and for any $(\gamma, \delta) \in R$, the output of $\widehat{V}(\gamma)$ after interacting with $\mathtt{P}(\delta)$ and that of $\mathtt{Sim}^{\widehat{V}(\gamma)}$ are identically distributed.

3.6 Our Scheme Overview

An oblivious transfer with fine grained access control from ciphertext policy attribute based encryption is run between the parties as follows: one credential issuer I, one database DB, and one or many users $\mathtt{U}_1, \cdots, \mathtt{U}_Z$, where Z is a positive integer. DB hosts a database $((m_1, \mathbb{A}_1), \cdots, (m_N, \mathbb{A}_N))$, where $m_l (l = 1, \cdots, N)$ is protected by access structure $\mathbb{A}_l (l = 1, \cdots, N)$. Each access structure $\mathbb{A}_l$ describes the attribute set that a user must possess to access m_l. Each user U possesses attribute set $S_\mathtt{U}$, and she can access messages m_l if and only if her $S_\mathtt{U}$ satisfies access structure $\mathbb{A}_l$. A credential issuer I certifies whether user U possesses attribute set $S_\mathtt{U}$.

The proposed scheme divides the interaction between parties into three phases as follows: A credential issuing phase, an initialization phase, and a transfer phase. In the credential issuing phase, a user U asks I to certify she has the attribute set $S_\mathtt{U}$. If certification succeeds, I issues U a credential on attributes $S_\mathtt{U}$. In the initialization phase, DB encrypts messages $m_l (l = 1, \cdots, N)$ under the corresponding access structure $\mathbb{A}_l (l = 1, \cdots, N)$, sends ciphertext $(C_1, \cdots, C_N)$ to each user U. In the transfer phase, the user U proves in zero-knowledge proof possession of a credential on her attribute set $S_\mathtt{U}$ to DB, and gets a

private key $PriKey_{S_{\mathtt{U}}}$ associated to her attribute set $S_{\mathtt{U}}$ from DB. If her $S_{\mathtt{U}}$ satisfies access structure $\mathbb{A}_l$, she can decrypt all the ciphertexts $C_l(l = 1, \cdots, N)$ to recover plaintext messages $m_l(l = 1, \cdots, N)$.

DB employs a ciphertext policy attribute based encryption scheme [19] to encrypt plaintext messages $(m_l)(l = 1, \cdots, N)$ under the corresponding access structure $\mathbb{A}_l(l = 1, \cdots, N)$. In a CP-ABE scheme, a ciphertext C is associated with access structure $\mathbb{A}$, whereas a private key $PriKey_S$ is associated with the user's attribute set S. If the attribute set S satisfies the access structure $\mathbb{A}$, then the private key $PriKey_S$ can decrypt the ciphertext C to recover plaintext message m.

In the transfer phase, the user U who possesses attribute set $S_{\mathtt{U}}$ can obtain a private key $PriKey_S$ of the CP-ABE scheme from the database DB. In traditional CP-ABE schemes, the Private Key Generator (PKG, for short) needs to learn the attribute set $S_{\mathtt{U}}$ to calculate a private key. Whereas the privacy properties in our scheme require the database acting as PKG should not learn the attribute set $S_{\mathtt{U}}$. Furthermore, DB assures that the user U only obtains the private keys associated with the attribute set $S_{\mathtt{U}}$. To handle these problems, we propose the expressive CP-ABE scheme with a blind key generation, where the user U proves in zero-knowledge proof possession of a credential on her attributes $S_{\mathtt{U}}$ to DB, and then she obtains a private key associated with $S_{\mathtt{U}}$ in a blind manner, such that DB does not learn $S_{\mathtt{U}}$.

We require authenticated communication between a user U and the issuer I, whereas communication between DB and U should be anonymous.

3.7 Credential Signature Scheme

The signature scheme comprises the following algorithms:

1) KeyGen (1^{κ}). The key generation algorithm takes in a security parameter κ, and outputs a keypair (sk, vk).

2) Sign $(sk, m_1, \cdots, m_N)$.The signing algorithm takes in a private signing key sk and one or more messages $m_1, \cdots, m_N$, and outputs the signature α.

3) Verify $(vk, \alpha, m_1, \cdots, m_N)$. The verification algorithm takes in a signature,messages(s) pair$(\alpha,(m_1, \cdots, m_N))$ and verification key vk, and outputs 1 if the signature verification is valid, 0 otherwise.

We extend a signature scheme with two protocols to achieve a credential scheme.First, a user U and a credential issuer I engage in an issuing protocol Issue by means of which U obtains a signature from I on a committed message $\mathcal{C}_{m_l}$ = Commit $(\mathbb{CP}, m_l,$ Decommit$)$, where $l = 1, \cdots, N$. Second, a protocol Show allows U to prove possession of a signature by I on a committed messages $\mathcal{C}_{m_l}$ to a verifier. To prevent users from colluding their credentials and to securely realize any credential scheme, we employ an ideal functionality $\mathtt{F}_{credential}$ as follows:

* On receiving (issue, att) from $\mathtt{U}_z(z = 1, \cdots, Z)$, where Z is a positive integer, and $att \in \mathbb{S}$, where $\mathbb{S}$ is the universe of attributes, it sends (issue, U, att) to I that sends back a bit β. If $\beta = 1$, then att is added to $S_{\mathtt{U}_z}$, and β is sent to $\mathtt{U}_z$; else β is simply sent to $\mathtt{U}_z$.

* On receiving (Show, $S^{\star}$) from $\mathtt{U}_z$, in which the cardinality of attribute set $S^{\star}$ is q, where q is a positive integer, if $S^{\star} \subseteq S_{\mathtt{U}}$, (verify, valid, q) is sent to $\mathtt{U}_z$ and to the verifier; else, (verify, invalid, q) is sent to $\mathtt{U}_z$ and to the verifier.

3.8 k-out-of-N Oblivious Transfer

An oblivious transfer scheme [3, 11] comprises four algorithms ($\mathtt{S}_I$, $\mathtt{R}_I$, $\mathtt{S}_T$, $\mathtt{R}_T$). In the initialization phase, an interactive protocol is run by the sender and the receiver. A state value S_0 is obtained by the sender via running $\mathtt{S}_I$ $(m_1, \cdots, m_N)$, and a state value R_0 is obtained by the receiver via running $\mathtt{R}_I$. Then, during the transfer phase, the sender and receiver interactively conduct $\mathtt{S}_T$, $\mathtt{R}_T$, respectively, k times as follows:

1) In the adaptive $\mathtt{OT}^N_{k\times 1}$ case, where $1 \leq l \leq k$, the l^{th} transfer proceeds as follows: the state value S_l is obtained by the sender via running $\mathtt{S}_T(S_{l-1})$, and the receiver runs $\mathtt{R}_T(R_{l-1}, \sigma_l)$ in which $1 \leq \sigma_l \leq N$ is the index of the message to be received. The receiver obtains state information R_l and the message $m^*_{\sigma_l}$ or $\perp$ which indicates protocol failure.

2) In the non-adaptive $\mathtt{OT}^N_k$ case, the parties conduct the protocol as in the aforementioned case. However, for each round $l < k$,the algorithm $\mathtt{R}_T(R_{l-1}, \sigma_l)$ does not return a message. At the end of the k^{th} transfer, $\mathtt{R}_T(R_{k-1}, \sigma_k)$ returns the messages $(m^*_{\sigma_1}, \cdots, m^*_{\sigma_N})$ in which for $l = 1, \cdots, N$, each $m^*_{\sigma_l}$ is a valid message or the symbol $\perp$ which indicates protocol failure. Our scheme employs k-out-of-N oblivious transfer realizing the ideal functionality $\mathtt{F}_{OT}$. The functionality $\mathtt{F}_{OT}$ performs as follows:

   * On receiving (Initialize, $(m_l, \mathbb{A}_l)_{l=1,\cdots,N}$) from DB, it sets DB $\leftarrow (m_l, \mathbb{A}_l)_{l=1,\cdots,N}$.

   * On receiving (transfer, $\sigma_1, \cdots, \sigma_k$) from $\mathtt{U}_z$, it proceeds as follows: It sends (receive, k) to DB. If DB is honest, $\mathtt{F}_{OT}$ sets $\{\beta_l = 1\}^k_{l=1}$, else, DB sends back (transfer, $\{\beta_l\}^k_{l=1}$). For $l = 1$ to k, if $\beta_l = 1$, $\mathtt{F}_{OT}$ sets $m^*_{\sigma_l} = m_{\sigma_l}$; else $\mathtt{F}_{OT}$ sends back $\perp$. $\mathtt{F}_{OT}$ returns (transfer, $m^*_{\sigma_1}, \cdots, m^*_{\sigma_k}$) to $\mathtt{U}_z$.

4 Blind Expressive Ciphertext Policy Attribute Based Encryption

4.1 Scheme Definition

Definition 1. *Ciphertext Policy Attribute Based Encryption (CP-ABE) scheme [19] comprises the four algorithms as follows:*

$ABE.Setup(1^k) \rightarrow (MS, PP)$: . It takes in a security parameter κ. It outputs a master secret MS employed to generate the users' private keys and the public parameters PP defining system attribute sets $\mathbb{S}$ which are employed by all parties in the scheme.

$ABE.Encrypt(PP, \mathbb{A}, m) \rightarrow \mathbf{CT}_{\mathbb{A}}$. It takes in the public parameters PP, the plaintext message m and the access structure $\mathbb{A}$ over a set of attributes specifying which users are able to decrypt to recover the plaintext message. It outputs the ciphertext $CT_{\mathbb{A}}$ associated with access structure $\mathbb{A}$.

$ABE.PriKeyGen(MS, S) \rightarrow PriKey_S$. It takes in the master secret MS, and the attribute set of user $S \subseteq \mathbb{S}$. It outputs the private key of user $PriKey_S$ associated with the attribute set of user S.

$ABE.Decrypt(CT_{\mathbb{A}}, PriKey_S) \rightarrow M$. It takes in the $CT_{\mathbb{A}}$ and the private key $PriKey_S$. It outputs the plaintext message m if attribute set S satisfies the access structures $\mathbb{A}$, else it returns $\perp$.

Correctness. *A CP-ABE scheme is correct when for all security parameters κ, all messages m, all sets of attributes S, access structures $\mathbb{A}$, all master secrets MS and public parameters PP output by ABE.Setup algorithm, all private keys $PriKey_S$ output by ABE.PriKeyGen algorithm, all ciphertexts $CT_{\mathbb{A}}$ output by ABE.Encrypt algorithm, if a set of attributes S satisfies access structure $\mathbb{A}$, the following proposition holds: $ABE.Decrypt(ABE.Encrypt(PP, \mathbb{A}, m), PriKey_s) = m$.*

4.2 Security Model for Ciphertext Policy Attribute Based Encryption Scheme

We describe a security model for CP-ABE scheme using a security game between a challenger and an attacker as follows:

Setup. The challenger runs the *ABE.Setup* algorithm which generates (MS, PP) and gives the attacker PP.

Phase 1. The attacker makes repeated private keys associated with attribute sets $S_1, \cdots, S_{Q_1}$, respectively.

Challenge. The attacker submits two plaintext messages m_0 and m_1 with $|m_0| = |m_1|$ and a challenge access structure $\mathbb{A}^*$ to the challenger with the restriction that none of the attribute sets $S_1, \cdots, S_{Q_1}$ from **Phase 1** satisfy the access structure $\mathbb{A}^*$. The challenger flips a random coin β, and encrypts m_β under $\mathbb{A}^*$. The resulting ciphertext $\mathtt{CT}^*$ is given to the attacker.

Phase 2. Phase 1 is repeated with the restriction that none of the attribute sets $S_{Q_1+1}, \cdots, S_Q$ satisfy the access structure $\mathbb{A}^*$ in the challenge phase.

Guess. The attacker outputs a guess $\beta^* \in \{0, 1\}$ of β, if $\beta^* = \beta$, the attacker wins.

Definition 2. *A CP-ABE scheme is secure against chosen plaintext attacks (CPA) if no probabilistic polynomial time attackers have non-negligible advantage in the aforementioned game, where the advantage is defined as $|Pr[\beta^* = \beta] - \frac{1}{2}|$.*

4.3 Blind Expressive Ciphertext Policy Attribute Based Encryption with Fine Grained Access Control

In the proposed oblivious transfer with fine grained access control (**AC-OT**) scheme, the database DB acts as `PKG`. When a user U who possesses attribute set $S_{\mathtt{U}}$ makes a request for DB, DB will check whether U possesses the credential of $S_{\mathtt{U}}$, if so, calculates $PriKey_{S_{\mathtt{U}}}$.

In traditional `CP-ABE` scheme due to [19], when a user U asks a private key associated with her attribute set $S_{\mathtt{U}}$, `PKG` will learn $S_{\mathtt{U}}$ to check whether U possesses the attributes $S_{\mathtt{U}}$, and calculate the private key $PriKey_{S_{\mathtt{U}}}$ by running $ABE.PriKeyGen$ algorithm. Whereas, in the proposed scheme, DB will accomplish the tasks without learning $S_{\mathtt{U}}$. To handle the problem, we extend traditional `CP-ABE` scheme with a blind private key generation protocol $ABE.BlindPriKeyGen$.

Definition 3. *If the underlying `CP-ABE` scheme (ABE.Setup, ABE.PriKeyGen, ABE.Encrypt, ASE.Decrypt) is secure and ABE.BlindPriKeyGen can be securely realized, then a blind `CP-ABE` scheme (ABE.Setup, ABE.BlindPriKeyGen, ABE.Encrypt, ABE.Decrypt) is secure.*

4.4 Additional Properties for a Blind CP-ABE Scheme

We employ blind `CP-ABE` scheme as a tool for constructing oblivious transfer with fine-grained access control.

Efficient POK for master secret. Our `AC-OT` constructions require an efficient zero-knowledge proof of knowledge protocol for the statement $\mathtt{POK}\{(msk) : (PP, msk) \in Setup(1^\kappa)\}$.

Committing CP-ABE scheme. To construct AC-OT protocols, we require our blind CP-ABE scheme should be committing.

The committing property requires that, given a ciphertext CT associated with an access structure $\mathbb{A}$, two different private keys associated with two different attribute sets satisfying $\mathbb{A}$ will yield a same plaintext message when the ciphertext CT is decrypted, which prevents a malicious database DB from calculating *mal-formed* ciphertexts, which makes the anonymity of the user be guaranteed. Two algorithms are defined as follows:

ABE.KeySanityCheck. This is a private key sanity check algorithm. It takes in public parameters PP, attribute set S, and private key $PriKey_S$ associated with S, and it outputs `Valid` if checks pass, else `Invalid`.

ABE.CiphertextSanityCheck. This is a ciphertext sanity check algorithm. It takes in public parameters PP and the ciphertext CT, and it outputs `Valid` if PP and the ciphertext CT are honestly generated, else `Invalid`.

Definition 4. *(Committing CP-ABE Scheme.) A (blind) CP-ABE scheme is committing if and only if: (1) It is secure according to Definition 1; (2) Each PPT attacker A has a negligible advantage in κ in the game as follows: First, A outputs public parameters PP, a ciphertext CT associated with access structure $\mathbb{A}$ and two different attribute sets S and S^* satisfying $\mathbb{A}$. If ABE.CiphertextSanityCheck outputs Invalid, then aborts, else the challenger runs the ABE.BlindPriKeyGen protocol with the attacker A twice on input (PP, S) and (PP, S^*) to obtain $PriKey_s$ and $PriKey_s^*$. The challenger runs ABE.KeySanityCheck(PP, S, $PriKey_S$) and ABE.KeySanityCheck(PP, S^*, $PriKey_{S^*}$) and aborts if the output of any of them is Invalid. The attacker A's advantage is defined to be: $|Pr[ABE.Decrypt(CT_{\mathbb{A}}, S, PriKey_S) \neq Pr[ABE.Decrypt(CT_{\mathbb{A}}, S^*, PriKey_{S^*})]|$.*

5 Scheme Construction

Blind expressive ciphertext policy attribute based encryption scheme employed to enforce fine-grained access control on the encrypted data is constructed as follows:

ABE.Setup$(1^k) \rightarrow (MS, PP)$. The setup algorithm calls the group generator algorithm $\mathbb{G}(1^\kappa)$ and obtains the descriptions of the two groups $\mathbb{G}$ and $\mathbb{G}_T$ and the bilinear map $\mathbb{D} = (p, \mathbb{G}, \mathbb{G}_T, g, e)$, in which p is the prime order of the cyclic groups $\mathbb{G}$ and $\mathbb{G}_T$, g is a generator of $\mathbb{G}$ and e is a bilinear map. The universe of system attributes are $\mathbb{S} = \{att_1, att_2, \cdots, att_{|\mathbb{S}|}\}$, where $|\mathbb{S}|$ is the cardinality of the universe $\mathbb{S}$ of system attributes. It selects the random exponents $t_1, t_2, \cdots, t_{|\mathbb{S}|}, \theta, \mu \in \mathbb{Z}_p^*$. For each attribute $att_d \in \mathbb{S}(1 \leq d \leq |\mathbb{S}|)$, it selects a corresponding $t_d \in \mathbb{Z}_p^*$, and sets $T_d = g^{t_d}(1 \leq d \leq |\mathbb{S}|)$. The public parameters are published as: $PP = (\mathbb{D}, g, e(g,g)^\mu, g^\theta, \{T_d\}_{1 \leq d \leq |\mathbb{S}|})$, where $e(g,g)^\mu$ can be pre-computed. The master secret is $MS = g^\mu$.

ABE.Encrypt$(PP, (M, \varphi), m) \rightarrow CT_{(M,\varphi)}$. The encryption algorithm encrypts a message $m \in \mathbb{G}_T$ under the access structure $\mathbb{A} = (M, \varphi)$, employing the public parameters PP. Let access matrix M be an matrix with h rows and n columns. The algorithm selects the column vector $\overrightarrow{v} = (s, x_2, \cdots, x_n)^T \in \mathbb{Z}_p^n$, where T is the transpose of the vector $\overrightarrow{v}$, $x_2, \cdots, x_n$ are uniformly at random chosen and $\overrightarrow{v}$ is employed to share the secret encryption exponent s. For any $j = 1, \cdots, h$, then $s_j = M_j \overrightarrow{v}$ is j^{th} share of the secret s according to Π, where M_j is the vector corresponding to the j^{th} row of M. Furthermore, the algorithm selects random elements $c_j \in \mathbb{Z}_p (j = 1, \cdots, h)$. The resulting ciphertext is constructed and calculated as follows:

$$\begin{aligned} CT_{(M,\varphi)} &= ((M,\varphi), E_b, E, \{E_j, F_j\}_{j=1,\cdots,h}). \\ E_b &= g^s \\ E &= m \cdot e(g,g)^{\mu s} \\ E_j &= g^{\theta s_j} T_{\varphi(j)}^{-c_j} \\ F_j &= g^{c_j}. \end{aligned}$$

PriKeyGen$(MS, S) \rightarrow PriKey_S$. The private key generation algorithm takes in the master secret MS and the attribute set of the user $S \subseteq \mathbb{S}$. For every user, it selects a random $r \in \mathbb{Z}_p^*$ employed to prevent collusion attacks through which the different users can pool their attributes to decrypt the ciphertext that they cannot decrypt individually and calculates the private key $PriKey_S$ as follows:

$$\begin{aligned} PriKey_S &= (K_b, D_b, \{K_d\}_{d \in S}). \\ K_b &= g^{\mu + \theta \cdot r} \\ D_b &= g^r \\ K_d &= T_d^r. \end{aligned}$$

ABE.Decrypt$(CT_{(M,\varphi)}, PriKey_S) \rightarrow m$. The decryption algorithm takes in $\mathrm{CT}_{(M,\varphi)}$ and $PriKey_S$. If attribute set S satisfies the access structure (M, φ), and let $J = \{j : j \in \{1, \cdots, h\} \wedge \varphi(j) \in S\}$. Then, there exist constants $\{\eta_j \in \mathbb{Z}_p\}_{j \in J}$ such that, if $\{s_j\}_{j \in J}$ are valid shares of a secret s according to M, then $\Pi_{j \in J} \eta_j s_j = s$. The decryption algorithm performs as follows:

Step 1. It calculates

$$\begin{aligned} V_1 &= \prod_{j\in J}((e(E_j, D_b)e(F_j, K_{\varphi(j)}))^{\eta_j} \\ &= \prod_{j\in J}(e(g^{\theta s_j}T_{\varphi(j)}^{-c_j}, g^r)e(g^{c_j}, T_{\varphi(j)}^r))^{\eta_j} \\ &= e(g,g)^{\sum_{j\in J}\theta r s_j \eta_j} \\ &= e(g,g)^{\theta r \sum_{j\in J} s_j \eta_j} \\ &= e(g,g)^{\theta rs}. \end{aligned}$$

Step 2. It calculates

$$\begin{aligned} V_2 &= e(E_b, K_b)/V_1 \\ &= e(g^s, g^{\mu+\theta\cdot r})/e(g,g)^{\theta rs} \\ &= e(g,g)^{\mu s}. \end{aligned}$$

Step 3. It calculates

$$\begin{aligned} E/V_2 &= m\cdot e(g,g)^{\mu s}/e(g,g)^{\mu s} \\ &= m. \end{aligned}$$

This scheme is provided with a zero-knowledge proof of knowledge of the statement $POK\{(MS) : (PP, MS) \in Setup(1^\kappa)\}$ that is given by $POK\{(\theta, g^\mu) : g^\theta \wedge e(g, g^\mu)\}$.

We prove that this **CP-ABE** scheme is committing as follows:

ABE.KeySanityCheck$(PP, S, PriKey_S)$**.** Parse $PriKey_S$ as $(K_b, D_b, \{K_d\}_{d\in S})$), and checks whether $e(K_b, g) = e(g^\theta, D_b)\cdot e(g,g)^\mu$ and for any attribute $d \in S$, $e(g, K_d) = e(D_b, T_d)$ holds. If so, it outputs `Valid`, else `Invalid`.

ABE.CiphertextSanityCheck$(PP, CT_{(M,\varphi)}$**.** Parse $\mathrm{CT}_{(M,\varphi)}$ as $((M,\varphi), E_b = g^s, E = m\cdot e(g,g)^{\mu s}$, $\{E_j = g^{\theta s_j}T_{\varphi(j)}^{-c_j}, F_j = g^{c_j}\}_{j=1,\cdots,h})$. Check whether $\prod_{\varphi(j)\in S} e(E_j, g)^{\eta_j} = \prod_{\varphi(j)\in S} e(F_j, T_{\varphi(j)}^{-1})^{\eta_j}\cdot e(g^\theta, E_b)$ holds. If so, output `Valid`; if not, output `Invalid`.

6 Blind Private Key Generation Protocol

A blind private key generation protocol is employed to extend **CP-ABE** scheme to enforce fine-grained access control on the encrypted data. Assuming database DB and credential issuer `I` operate on a universe $\mathbb{S}$ of attribute. U obtains a credential certifying that U has attribute set S from `I`. U and `I` engage in the credential issuing protocol as follows.

Issue():

1) U obtains $S_U \leftarrow S_U \cup \{att\}$ and sends S_U to I.
2) I checks S_U.
3) I generates a credential for S_U, i.e. $Cred(S_U)$, and sends it to U.
4) U obtains $Cred(S_U)$.

We employ full simulatable k-out-of-N oblivious transfer scheme and the credential scheme.`I` runs KeyGen(1^κ) of the credential scheme to generate $(PriKey_{\mathtt{I}}, PK_{\mathtt{I}})$. DB runs $ABE.Setup(1^k)$ to generate PP, MS. U have both $PK_{\mathtt{I}}$ and PP. A blind private key generation protocol for **CP-ABE** scheme is depicted as follows.

BlindPriKeyGen():

1) U calculates commitments $\{(Com_i, Decom_i) = Commit(i)\}_{i\in S}$ and sends $\{Com_i\}_{i\in S}$ to DB.
2) U proves in zero-knowledge possession of credential $Cred(S_U)$ to DB.
3) If the proof fails, abort.
4) DB runs $PriKeyGen(MS, S) \rightarrow PriKey_S$ to generate $PriKey_S = (K_b = g^{\mu+\theta\cdot r}, D_b = g^r, \{K_d = T_d^r\}_{d\in S})$. DB as a sender and U as a receiver runs a full simulatable k-out-of-N oblivious transfer protocol.
5) U inputs S as selection values.
6) DB inputs commitments $\{com_i\}_{i\in S}$ and $\{K_d\}_{d\in\mathbb{S}}$ as messages to be received.
7) U obtains $\{K_d\}_{d\in S}$.
8) DB returns nothing.
9) DB sends K_b, D_b to U.
10) U sets $PriKey_S = (K_b = g^{\mu+\theta\cdot r}, D_b = g^r, \{K_d = T_d^r\}_{d\in S})$ and calls $ABE.KeySanityCheck(PP, S, PriKey_S)$, if the output is `Valid`, U obtains PriKey_S.

As a result, U obtains a private key associated with S and DB does not learn anything about S.

Theorem 1. *This Blind Private Key Generation protocol can securely realize F_{BPKG}.*

Proof. We define a simulator $\mathtt{Sim}_{BPKG}$ which runs `A` as a subroutine and interacts with $\mathtt{F}_{BPKG}$. Given a real world attacker `A`, we construct an ideal world attacker $\mathtt{A}^*$ such that no environment `E` can distinguish between the real and the ideal world. Security is proved under a secure credential scheme and a secure oblivious transfer scheme. The secure credential scheme implies a simulator $Sim_{Credential}$ which interacts with $\mathtt{F}_{Credential}$ and `E` such that `E` cannot distinguish between the real and the ideal world. The secure oblivious transfer scheme implies a simulator Sim_{OT} which interacts with $\mathtt{F}_{OT}$ and `E` such that `E` cannot distinguish the real world from the ideal world.

The cases are not considered as follows: All parties are honest, all parties are dishonest, the issuer is the only honest party, or the issuer is the only dishonest party, since theses cases have no real practical interest.

For the remaining each case, we define a sequence of games to prove the indistinguishability between the real and ideal worlds. Let $\mathtt{Adv}[\mathbf{Game\ I}]$ denotes the advantage that $\mathtt{E}$ distinguishes between the ensemble of **Game I** and that of the real execution. We consider the four cases as follows. □

Case 1. *When attacker $\mathtt{A}$ corrupts the issuer $\mathtt{I}$ and the database DB, the ensembles $\boldsymbol{REAL}_{A,E}$ and $\boldsymbol{IDEAL}_{A^*,E}$ are computationally indistinguishable provided that the credential scheme is secure and the oblivious transfer scheme is secure.*

Proof. By applying all the changes represented in $Sim_{Credential}$, the environment $\mathtt{E}$ cannot distinguish between the real world and the ideal world provided that the credential scheme is secure. By applying all the changes represented in Sim_{OT}, the environment $\mathtt{E}$ cannot distinguish between the real world and the ideal world, provided that the oblivious transfer scheme is secure. Therefore, this distribution is identical to that of Sim_{BPKG}. □

Case 2. *When attacker $\mathtt{A}$ corrupts the database DB, the ensembles $\boldsymbol{REAL}_{A,E}$ and $\boldsymbol{IDEAL}_{A^*,E}$ are computationally indistinguishable provided that the credential scheme is secure and the oblivious transfer scheme is secure.*

Proof. The proof is similar to that of Case 1. □

Case 3. *When attacker $\mathtt{A}$ corrupts some users, the ensembles $\boldsymbol{REAL}_{A,E}$ and $\boldsymbol{IDEAL}_{A^*,E}$ are computationally indistinguishable, provided that the credential scheme is secure,the oblivious transfer scheme is secure and the commitment scheme is binding.*

Proof. By applying all the changes represented in $Sim_{Credential}$, the environment $\mathtt{E}$ cannot distinguish between the real world and the ideal world, provided that the credential scheme is secure. By applying all the changes represented in Sim_{OT}, the environment $\mathtt{E}$ cannot distinguish between the real world and the ideal world, provided that the oblivious transfer scheme is secure. If the selection values $\sigma_1, \cdots, \sigma_k$ generated by Sim_{OT} is different from attribute sets S generated by Sim_{OT}, it means that $\mathtt{A}$ can de-commit any of the commitments to two different values, which happens with negligible probability since the commitment scheme is binding. Therefore, this distribution is identical to that of Sim_{BPKG}. □

Case 4. *When attacker $\mathtt{A}$ corrupts the issuer $\mathtt{I}$ and some users U, the ensembles $\boldsymbol{REAL}_{A,E}$ and $\boldsymbol{IDEAL}_{A^*,E}$ are computationally indistinguishable provided that the credential scheme is secure, the oblivious transfer scheme is secure and the commitment scheme is binding.*

Proof. The proof is similar to that of Case 3. □

7 Fully Simulatable Oblivious Transfer with Fine Grained Access Control

Definition 5. *(Functionality $F_{SOTFGAC}$) Functionality $F_{SOTFGAC}$ performs as follows:*

* *On receiving (`issue`, att) from $\mathtt{U}_z$, in which att $\in \mathbb{S}$, it sends (issue, $\mathtt{U}_z$, att) to $\mathtt{I}$ that sends back a bit β. If $\beta = 1$, then att is added to $S_{\mathtt{U}_z}$, and β is sent to $\mathtt{U}_z$; else β is simply sent to $\mathtt{U}_z$.*
* *On receiving (`initialize`, $(m_l, \mathbb{A}_l)_{l=1,\cdots,N}$) from $\mathtt{DB}$, it sets $DB \leftarrow (m_l, \mathbb{A}_l)_{l=1,\cdots,N}$.*
* *On receiving (`transfer`, S) from $\mathtt{U}_z$, it proceeds as follows: If $DB \neq \perp$, it sends `transfer` to DB that returns a bit β. If $\beta = 0$ or $DB = \perp$, it sends $\perp$ to $\mathtt{U}_z$. If $\beta = 1$ and S satisfies access structures $\mathbb{A}_l$, it sends $(m_1^*, \cdots, m_N^*)$ to $\mathtt{U}_z$.*

7.1 Construction

Our construction employs the blind **CP-ABE** scheme to certify the attributes of users and to enforce fine grained access control. Here, $m_1, \cdots, m_N \in \{0,1\}^n$, and hash functions $H : m \rightarrow \{0,1\}^n$ are modelled as a random oracle. Credential Issuing phase is depicted in Section 6. Initialization phase is depicted as follows:

1) DB_I: Select $(PP, MS) \leftarrow ABE.Setup(1^k)$.
2) DB_I: Select random values $W_l \in \mathbb{G}_T$, and for $l = 1, \cdots, N$ set:
$$\begin{aligned} A_l &= ABE.Encrypt(PP, \mathbb{A}_l, W_l) \\ B_l &= H(W_l) \bigoplus m_l \\ C_l &= (A_l, B_l). \end{aligned}$$
3) DB_I: Execute $PoK\{(MS): (PP, MS) \in ABE.Setup(1^k)\}$.
4) DB_I: Send $\{PP, C_1, \cdots, C_N\}$ to U.
5) U_I: If the proof does not verify, or $ABE.CiphertextSanityCheck$ returns `Invalid`, these ciphertexts are rejected.
6) U_I: $R_0 = (PP, (C_1, \cdots, C_N))$.
7) DB_I: Return $S_0 = (PP, MS)$.

In the l^{th} transfer, `BlindPriKeyGen()` is run, and U obtains PriKey_S. Transfer phase is depicted as follows:

1) U_T: If `BlindPriKeyGen()` fails, then $m_l^* = \perp$.
2) U_T: Else for $l = 1$ to N, U_T checks whether S satisfies the access structure $\mathbb{A}_l$. If so, U runs $W_{\sigma_l} = ABE.Decrypt(PP, A_{\sigma_l}, PriKey_S)$, and obtains the messages $m_{\sigma_l}^* = H(W_{\sigma_l}) \bigoplus B_{\sigma_l}$; otherwise, $m_{\sigma_l}^* = \perp$.

3) U_T: Return $R_l = (R_{l-1}, m^*_{\sigma_l}, S)$.

4) DB_T: Return $S_l = (S_{l-1})$.

The encryption technique employed is secure in the random oracle model.

7.2 Proof of Security

Theorem 2. *$F_{SOTFGAC}$ is securely realized by fully simulatable oblivious transfer with fine grained access control.*

Proof. Given a real world attacker A, we construct an ideal world attacker $\mathtt{A}^*$ such that no environment E can distinguish between the real and the ideal world.The cases are not considered as follows: all parties are honest, all parties are dishonest, the issuer is the only honest party, or the issuer is the only dishonest party, since theses cases have no real practical interest.

For the remaining each case, we define a sequence of games to prove the indistinguishability between the real world and the ideal world. Let Adv[**Game I**] denotes the advantage that E distinguishes between the ensemble of **Game I** and that of the real execution. We define some set of negligible functions in which $\upsilon_n()$ denotes the n^{th} function. We consider the four cases as follows. □

Case 1. *When the real world attacker A corrupts I and DB, the ensembles $\boldsymbol{REAL}_{A,E}$ and $\boldsymbol{IDEAL}_{A^*,E}$ are computationally indistinguishable provided that proofs of knowledge are extractable, the **CP-ABE** is committing and secure is the blind private key generation with access control in the random oracle model.*

Proof.

Game 0. In this game, the dishonest real world DB and I interacts with the real world honest users $\mathtt{U}_z$. Hence, $Adv[Game0] = 0$.

Game 1. This game is the same as Game 0, except that the knowledge extractor is employed to extract MS from the proof of knowledge $\mathtt{POK}\{MS : (PP, MS) \in ABE.Setup(1^k)\}$. Since this extractor succeeds with all but negligible probability, Adv[Game 1] − Adv[Game 0] $\leq \upsilon_1(\kappa)$.

Game 2. This game is the same as Game 1, except that all ill-formed ciphertexts detected by running $ABE.CiphertextSanityCheck$ are rejected. The committing CP-ABE scheme ensures that if a ciphertext is valid via $ABE.CiphertextSanityCheck$, then it decrypts to the same message by employing any valid private key.Hence, Adv[Game 2]−Adv[Game 1] = 0.

Game 3. This game is the same as Game 2, except that it carries out all the changes depicted in $\mathtt{Sim}_{BPKG}$. Since the blind private key generation with fine grained access control is secure, it holds that Adv[Game 3] − Adv[Game 2] $\leq \upsilon_3(\kappa)$.

By summation, it holds that Adv[Game 3]−Adv[Game 0] = Adv[Game 3] is negligible. □

Case 2. *When the real world attacker A corrupts DB, the ensembles $\boldsymbol{REAL}_{A,E}$ and $\boldsymbol{IDEAL}_{A^*,E}$ are computationally indistinguishable provided that secure is the blind private key generation with access control in the random oracle model,proofs of knowledge are extractable, and the **CP-ABE** is committing.*

Proof. This proof is similar to that of Case 1. □

Case 3. *When the real world attacker A corrupts some users $\mathtt{U}_z$, the ensembles $\boldsymbol{REAL}_{A,E}$ and $\boldsymbol{IDEAL}_{A^*,E}$ are computationally indistinguishable provided that secure is the blind private key generation with access control in the random oracle model, proofs of knowledge are zero knowledge, and the **CP-ABE** scheme is secure.*

Proof.

Game 0. In this game, the honest real world DB and I interacts with the real world cheating user U . Hence, $Adv[Game0] = 0$.

Game 1. This game is the same as Game 0, except that a simulated proof is employed to replace the proof of knowledge $\mathtt{POK}\{MS : (PP, MS) \in ABE.Setup(1^k)\}$. Based on the zero-knowledge property of the zero-knowledge proof, it holds that Adv[Game 1] − Adv[Game 0] $\leq \upsilon_1(\kappa)$.

Game 2. This game is the same as Game 1, except that we employ all the changes presented in $\mathtt{Sim}_{BPKG}$. The secure blind private key generation with access control protocol implies that protocol execution in the real world is indistinguishable from the interaction between $\mathtt{Sim}_{BPKG}$ and $\mathtt{F}_{BPKGAC}$. Hence, Adv[Game 2] − Adv[Game 1] $\leq \upsilon_2(\kappa)$.

Game 3. This game is the same as Game 2, except that random values are employed to replace $B_1, \cdots, B_N$. We construct an algorithm A which breaks the security of the CP-ABE with non-negligible advantage. The challenger of the security game of the CP-ABE gives the public parameters of the CP-ABE to A. For $l = 1$ to N, A selects a random P_0,sets $P_1 = A_l$ and sends (P_0, P_1) to the challenger. The challenger tosses a fairly binary coin β and sends back a challenge ciphertext $A_l = ABE.Encrypt(PP, \mathbb{A}_l, P_\beta)$. A continues the simulation. On receiving a query, if it is not equal to P_0, or P_1 then A returns $\perp$. If it is P_0, A sets $\beta^* = 0$, else, A sets $\beta^* = 1$. A sends β^* to the challenger. The distribution in Game 3 is the same as that of the simulation. Hence, it holds that Adv[Game 3] − Adv[Game 2] $\leq \upsilon_3(\kappa)$.

By summation, it holds that Adv[Game 3]−Adv[Game 0] = Adv[Game 3] is negligible. □

Case 4. *When the real world attacker* $\mathtt{A}$ *corrupts some users* U *and* $\mathtt{I}$*, the ensembles* $\boldsymbol{REAL}_{A,E}$ *and* $\boldsymbol{IDEAL}_{A^*,E}$ *are computationally indistinguishable provided that secure is the blind private key generation with access control in the random oracle model, proofs of knowledge are zero knowledge, and the* ***CP-ABE*** *scheme is secure.*

Proof. The proof of Case 4 is similar with that of Case 3. □

8 Performance Evaluation

As depicted in Table 1 where *cat* denotes *category* and || denotes the cardinality of the set: for access policy, CDN scheme supports conjunction, and disjunction via duplication, whereas our scheme supports conjunction, disjunction and threshold directly. For the encrypted record size, given a conjunction normal form $(I_{1,1} \vee \cdots \vee I_{1,y_1}) \wedge \cdots \wedge (I_{n,1} \vee \cdots \vee I_{n,y_n})$, we represent it by employing an access tree whose internal nodes are `OR` gates and `AND` gates, and leaf nodes denote attributes; in our scheme, the encrypted record size is $\sum_{i=1}^{n} y_i$, whereas, in CDN scheme, the encrypted record size is $\prod_{i=1}^{n} y_i$ due to disjunction via duplication. By directly supporting disjunction, our scheme greatly reduces the size of encrypted database. For credential issuing phase, both schemes have communication complexity linear in the number of attributes possessed by some user. For initialization phase, both schemes have computation complexity linear in the number N of messages. For transfer phase, our scheme have communication complexity linear in the cardinality of the attribute universe, whereas that of CDN scheme is linear in the attribute number possessed by some user;however, in the CDN scheme, user U only obtains a record, whereas our scheme U obtains all the records which she is authorized to access. Hence, when fine grained access control needs to be enforced or the number of records authorized to access is larger, our scheme is more efficient than the CDN scheme.

9 Conclusion and Future Work

In this work, we propose an oblivious transfer scheme with fine grained access control from ciphertext policy attribute based encryption which greatly enhances expressiveness for access policies, and reduces the size of encrypted database. Furthermore, the communication complexity in the transfer phase of our scheme is constant in the number of records accessed. In the future work, we will design a scheme where access policies are hidden to furthermore enhance privacy.

10 Acknowledgments

This work was supported by the National Science Foundation of China (No. 61402376). The authors gratefully acknowledge the anonymous reviewers for their valuable comments.

References

[1] A. Beimel, *Secure Schemes for Secret Sharing and Key Distribution*, Ph.D Thesis, Israel Institute of Technology, Technion, Haifa, Israel, 1996.

[2] J. Bethencourt, A. Sahai, and B. Waters, "Ciphertext-policy attribute-based encryption," in *IEEE Symposium on Security and Privacy*, pp. 321–334, 2007.

[3] J. Camenisch, M. Dubovitskaya, and G. Neven, "Oblivious transfer with access control," in *Proceedings of the 16th ACM Conference on Computer and Communications Security*, pp. 131–140, 2009.

[4] T. Y. Chang and W. P. Yang M. S. Hwang, "An improved multi-stage secret sharing scheme based on the factorization problem," *Information Technology and Control*, vol. 40, pp. 246–251, 2011.

[5] L. Cheung and C. C. Newport, "Provably secure ciphertext policy ABE," in *ACM Conference on Computer and Communications Security*, pp. 456–465, 2007.

[6] P. S. Chung, C. W. Liu, and M. S. Hwang, "A study of attribute-based proxy re-encryption scheme in cloud environments," *International Journal of Network Security*, vol. 16, no. 1, pp. 1–13, Jan. 2014.

[7] S. Coully, M. Green, and S. Hohenberger, "Controlling access to an oblivious database using stateful anonymous credentials," in *Cryptology ePrint Archive, Report 2008/474*, 2008.

[8] R. Cramer, I. Damgard, and P. D. MacKenzie, "Efficient zero-knowledge proofs of knowledge without intractability assumptions," in *3rd International Workshop on Theory and Practice in Public Key Cryptography (PKC'00)*, LNCS 1751, pp. 354–372, Melbourne, Victoria, Australia, Jan. 2000.

[9] V. Goyal, A. Jain, O. Pandey, and A. Sahai, "Bounded ciphertext policy attribute-based encryption," in *Proceedings of 35th International Colloquium (ICALP'08)*, pp. 579–591, Reykjavik, Iceland, July 7-11, 2008.

[10] V. Goyal, O. Pandey, A. Sahai, and B. Waters, "Attribute-based encryption for fine-grained access control of encrypted data," in *Proceedings of the 13th ACM Conference on Computer and Communications Security*, pp. 89–98, 2006.

[11] M. Green and S. Hohenberger, "Blind identity-based encryption and simulatable oblivious transfer," in *Advances in Cryptology (ASIACRYPT'07)*, LNCS 4833, pp. 265–282, 2007.

[12] C. C. Lee, P. S. Chung, and M. S. Hwang, "A survey on attribute-based encryption schemes of access control in cloud environments," *International Journal of Network Security*, vol. 15, no. 4, pp. 231–240, July 2013.

Table 1: Comparison of our scheme with CDN scheme

References	Access Policy	Encrypted Record Size	Credential Issuing Phase	Initialization Phase	Transfer Phase
CDN Scheme [3]	conjunction and disjunction via duplication	$\prod_{i=1}^{n} y_i$	$\mathcal{O}(\|cat\|)$	$\mathcal{O}(N)$	$\mathcal{O}(\|cat\|)$
Our Scheme	conjunction, disjunction and threshold	$\sum_{i=1}^{n} y_i$	$\mathcal{O}(\|S\|)$	$\mathcal{O}(N)$	$\mathcal{O}(\|\mathbb{S}\|)$

[13] A. Lewko, T. Okamoto, A. Sahai, and K. Takashima, "Fully secure functional encryption: Attribute-based encryption and (hierarchical) inner product encryption," in *Advances in Cryptology (EUROCRYPT'10)*, LNCS 6110, pp. 62–91, 2010.

[14] T. Pedersen, "Non-interactive and information-theoretic secure verifiable secret sharing," in *Advances in Cryptology (CRYPTO'91)*, LNCS 576, pp. 129–140, Heidelberg, 1991.

[15] M. Pirretti, P. Traynor, P. McDaniel, and B. Waters, "Secure attribute-based systems," in *ACM Conference on Computer and Communications Security*, pp. 99–112, 2006.

[16] A. S. R. Ostrovsky and B. Waters, "Attribute-based encryption with non-monotonic access structures," in *ACM Conference on Computer and Communications Security*, pp. 195–203, 2007.

[17] A. Sahai and B. Waters, "Fuzzy identity based encryption," in *Advances in Cryptology (Eurocrypt'05)*, LNCS 3494, pp. 457–473, 2005.

[18] Subba Rao Y V and Chakravarthy Bhagvati, "CRT based threshold multi secret sharing scheme," *International Journal of Network Security*, vol. 16, no. 4, pp. 249–255, 2014.

[19] B. Waters, "Ciphertext-policy attribute-based encryption: An expressive, efficient, and provably secure realization," in *Public Key Cryptography (PKC'11)*, LNCS 6571, pp. 53–70, 2011.

[20] Ye Zhang, M. Ho Au, D. S. Wong, Q. Huang, N. Mamoulis, D. W. Cheung, and S. M. Yiu, "Oblivious transfer with access control: Realizing disjunction without duplication," *Pairing-Based Cryptography (Pairing'10)*, LNCS 6487, pp. 96–115, 2010.

Xingbing Fu is a lecturer, he received his M.S. degree from Southwest University in 2007. He is currently a PhD Candidate in the School of Computer Science and Engineering, University of Electronic Science and Technology of China. His research interests are information security, cloud computing, and cryptography.

Shengke Zeng is a lecturer at the School of Mathematics and Computer Engineering, Xihua University. She received her Ph.D. degree from University of Electronic Science and Technology of China in 2013. Her research interests include: Cryptography and Network Security.

Fagen Li received the Ph.D. degree in Cryptography from Xidian University, Xi'an, P.R. China in 2007. His research interests include cryptography and network security.

Anomaly Detection Using an MMPP-based GLRT

Chris Scheper[1] and William J. J. Roberts[2]
(Corresponding author: Chris Scheper)

AthenaHealth, Inc.[1]
311 Arsenal Street, Watertown, MA 02472, USA
(Email: cjscheper@gmail.com)
Opera Solutions, LLC[2]
12230 El Camino Real, San Diego, CA 92116, USA
(Received Apr. 3, 2013; revised and accepted Jan. 10 & Mar. 13, 2014)

Abstract

Detection of anomalous network traffic is accomplished using a generalized likelihood ratio test (GLRT) applied to traffic arrival times. The network traffic arrival times are modelled using a Markov modulated Poisson process (MMPP). The GLRT is implemented using an estimate of the MMPP parameter obtained from training data that is not anomalous. MMPP parameter estimation is accomplished using Rydén's expectation-maximization (EM) approach. Using data from the 1999 DARPA intrusion detection evaluation, the performance of a GLRT using an MMPP, a Poisson process, and a mixture of exponentials is compared. The MMPP-based GLRT has the best performance and the largest computational requirements.

Keywords: Anomaly detection, generalized likelihood ratio test, markov-modulated Poisson process

1 Introduction

Anomaly detection using network packet arrival times is a binary hypothesis testing problem. Let H_1 denote the hypothesis that the network is receiving packets with anomalous arrival times. Let H_0 denote the hypothesis that network packet arrival times are not anomalous. If the probability density functions (pdfs) of the arrival times under the two hypotheses are known, then optimum decision rule in the Neyman-Pearson sense is given by the likelihood ratio test (pp. 32, Theorem 1) [14]. The true pdfs, however, are not generally known. In the "plug-in" approach, a parametric form for the pdfs is prescribed, the parameters are estimated from training signals, and the resulting pdfs are used in the likelihood ratio test as if they were the true pdfs. For network anomaly detection, although training signals for H_0 may be available, appropriate training signals for H_1 are generally difficult to obtain for a number of reasons. Anomalies are generally difficult to characterize. An intruder attempting to gain unauthorized network access may enjoy greater success using an approach that is unknown to network security systems. In this paper, we apply the generalized likelihood ratio test (GLRT) [26] to anomaly detection. The GLRT does not require an explicit pdf for H_1. Instead, a parametric form for this pdf is assumed and the parameter is estimated from the *test* signal. The GLRT is widely applied in signal classification problems, see e.g. [1, 13, 25]. Optimality of the GLRT is discussed in [28]. Discussions of the characterization of normal behvavior H_0 can be found in [12, 29].

The Markov modulated Poisson process (MMPP) constitutes the pdf we prescribe for network arrival times. The MMPP is a conditional Poisson process whose intensity is controlled by a Markov chain. A summary of the properties of MMPPs can be found in [3]. Other MMPP applications include modeling rainfall, pollution, minke whale observations, photon arrivals due to fluorescence, financial defaults, fraud in banking, and target tracking, see, e.g., [27] and the references therein. Heffes [7] in 1980 and Heffes and Lucantoni [8] in 1986 established that the MMPP faithfully models key properties of Internet traffic, including the mean arrival rate and the variance-to-mean ratio. The MMPP has subsequently become well established as a model for Internet traffic with numerous references in the literature, see, e.g., [9, 17, 23] for recent examples. Studies of anomaly detection in network traffic using other models can be found in [4, 15, 24]. As Internet applications can involve very large amounts of data, and due to its desirable asymptotic properties, we aim for a maximum likelihood (ML) estimate of the MMPP parameter. There is no explicit form for the ML MMPP parameter estimate. Instead, a number of expectation maximization (EM) approaches have been proposed. In [21], Rydén developed an EM algorithm that, in contrast to previous algorithms [2], had explicit ex-

pectation steps and maximization steps. Computational aspects of Rydén's algorithm were improved by [19].

The MMPP has been previously applied to anomaly detection, but not, to our knowledge, as part of a GLRT. Detection of fraudulent intrusions on a telephone network was investigated by Scott [22] using an 2-state MMPP where one state represented a valid call and the other state represented a fraudulent call. Gibbs sampling was applied for parameter estimation. An anomaly was declared if the posterior probability of the fraud state was greater than a threshold. Ihler, Hutchins, and Smyth [11] applied an MMPP to anomalous event detection in freeway traffic and in building entry data. The MMPP parameter was estimated using Markov chain Monte-Carlo techniques and events were detected using their posterior probability. Pawling *et al.* [18] investigated detection of emergencies and natural disasters using the Kolmogorov-Smirnov test to compare simulated MMPP data to real cellular communication data. The MMPP parameter was estimated using a clustering algorithm.

The remainder of this paper is organized as follows. In Section 2, we formulate the GLRT for anomaly detection. In Section 3, we describe Rydén's EM algorithm for MMPP parameter estimation with the computational improvements of [19]. In Section 4, we describe numerical experiments performed using data from the 1999 DARPA intrusion detection evaluation [6]. In Section 5, we provide some concluding comments.

2 Binary Hypothesis Testing

Let $Y^n = \{Y_1, \ldots, Y_n\}$ denote a sequence of n positive random variables representing network packet interarrival times. Let $y^n = \{y_1, \ldots y_n\}$ denote a realization of Y^n. Let $p(y^n; \phi)$ denote an assumed parametric form of the pdf of Y^n, where ϕ is the parameter. Let ϕ_0 denote the parameter corresponding to network traffic that is not anomalous. Anomaly detection is to chose which of the following two hypotheses is true

$$\begin{aligned} H_0: &\quad y^n \sim p(y^n; \phi_0), \\ H_1: &\quad y^n \sim p(y^n; \phi) \text{ where } \phi \neq \phi_0. \end{aligned}$$

In statistical parlance, this is a classification problem for one simple and one composite hypothesis [14]. A hypothesis is called *simple* if the signal is described by a known pdf. A hypothesis is called *composite* if the pdf of the signal is only known to be a member of a family of pdfs. If ϕ is assumed random with a known pdf, the composite hypothesis can be represented as a simple hypothesis using a Bayesian approach, see, e.g., [20]. Here we adopt an approach based on the GLRT [26]. In this form of the GLRT, the unknown parameter of the process under the composite hypothesis is estimated in ML sense from the test signal, and used as if it were the correct parameter. The GLRT test statistic is given by

$$\delta(y^n; \phi_0) = \frac{p(y^n; \phi_0)}{\max_\phi p(y^n; \phi)}, \tag{1}$$

and the decision is made according to

$$\frac{1}{n} \log \delta(y^n; \phi_0) \underset{H_1}{\overset{H_0}{\gtrless}} \eta \tag{2}$$

where η is a threshold. The GLRT does not require knowledge of the parameter corresponding to H_1. It does, however, require an explicit ϕ_0 which may be estimated from training signals obtained when the network is not under attack.

Asymptotic optimality of a GLRT in the Neyman-Pearson sense was shown for independent identically distributed (iid) sources [10] and Markov chain sources of any given order [5, 28, 30]. Optimality of an extension of the GLRT to model order estimation was shown in [16]. Although optimality of the GLRT has not been shown for the processes we consider, the GLRT is widely applied in other applications, see e.g. [1, 13, 25], where optimality also cannot be shown.

There are two events useful for characterizing performance of the GLRT: a *false alarm*, i.e., choosing H_1 when H_0 is true and a *detection*, i.e., choosing H_1 when H_1 is true. The loci of the probabilities of these events for various thresholds η is termed a receiver operator characteristic (ROC) curve. Generally, it is the relative frequencies of these events obtained from known test signals that are plotted.

3 MMPP Description and Estimation

An MMPP is a conditional Poisson process whose intensity is determined by an underlying continuous-time Markov chain. Let $\{N(t), t > 0\}$ denote the observed conditional Poisson process and let $\{X(t), t \geq 0\}$ denote the underlying continuous-time Markov chain with a state space $\{1, \ldots, r\}$. Let the $r \times r$ matrix Q denote the generator matrix of $X(t)$. Let π denote a $1 \times r$ vector of initial state probabilities of $X(t)$. Let the intensity of the conditional Poisson process at time t be given by λ_i when $X(t) = i$. Let Λ be the $r \times r$ diagonal matrix with diagonal elements given by $\{\lambda_i\}$. Generally, the expressions that we consider involve the sequence of event interarrival times Y^n, so that the event count $N(t)$ is given by $N(t) = \max\{j \mid \sum_{i=0}^{j} Y_i \leq t\}$, where $Y_0 = 0$. Generically, the role of π diminishes as $t \to \infty$. Therefore, we define the MMPP parameter of interest as $\phi = \{Q, \Lambda\}$.

Let $\mathbf{1}$ denote a $r \times 1$ vector of ones. The MMPP pdf is given by

$$p(y^n; \phi) = \pi \prod_{t=1}^{n} f(y_t; \phi) \mathbf{1},$$

where $f(y_t; \phi)$ represents the MMPP transition density matrix

$$f(y_t; \phi) = \exp((Q - \Lambda) y_t) \Lambda.$$

Considering y^n as training signals, we aim to find an ML

estimate

$$\hat{\phi} = \arg\max_{\phi} p(y^n; \phi)$$

There is no explicit form for the ML estimate, therefore we resort to Rydén's MMPP EM algorithm using computational improvements suggested in [19]. Here we provide only the details necessary for implementation; the full derivations are available in [19, 21]. Let $\phi = \{Q, \Lambda\}$ denote an existing parameter estimate. The first step is to recursively calculate the $1 \times r$ vectors of forward densities $\{L(t)\}$ and $r \times 1$ vectors of backward densities $\{R(t)\}$. Define $L(0) = \pi$ and $R(k+1) = \mathbf{1}$. The scaled recursions are given by

$$L(t) = \frac{L(t-1)f(y_t)}{c_t}, \quad R(t) = \frac{f(y_t)R(t+1)}{c_t}, \qquad (3)$$

where the scaling factor c_t is given by

$$c_t = L(t-1)f(y_t)\mathbf{1}. \qquad (4)$$

The log-likelihood of y^n can be readily calculated using

$$\log p(y^n; \phi) = \sum_{t=1}^{n} \log c_t. \qquad (5)$$

Given the forward and backward densities, we can then calculate the $r \times 1$ vector M and the $2r \times 2r$ matrices $\{C_t\}$ given by

$$M = \sum_{t=1}^{n} L(t)' \odot R(t+1),$$

where $\odot$ denotes element-wise multiplication and

$$C_t = \begin{bmatrix} Q - \Lambda & \Lambda R(t+1)L(t-1) \\ 0 & Q - \Lambda \end{bmatrix}$$

Denote by $\mathcal{I}_t$ the upper-right $r \times r$ block of the matrix exponential $e^{C_t y_t}$, and let $m = Q \odot \sum_{t=1}^{n} \mathcal{I}_t / c_t$ The updated estimates $\hat{\phi} = \{\hat{Q}, \hat{\Lambda}\}$ are calculated using

$$\hat{\lambda}_i = \frac{q_{ii} M_i}{m_{ii}},$$
$$\hat{q}_{ij} = \frac{q_{ii} m_{ij}}{m_{ii}}, \quad i \neq j.$$

The diagonal elements of $\hat{Q}$ are set so the rows of $\hat{Q}$ sum to zero. Let $\{\hat{\phi}^k\} = \{(\hat{Q}^k, \hat{\Lambda}^k)\}$ denote a sequence of estimates resulting from the iteration of this procedure. The EM algorithm guarantees that $p(y^n; \hat{\phi}^k) \geq p(y^n; \hat{\phi}^{k-1})$. The EM algorithm is terminated when the following convergence criterion is satisfied

$$\log p(y^n; \hat{\phi}^k) - \log p(y^n; \hat{\phi}^{k-1}) < \epsilon \qquad (6)$$

for $\epsilon > 0$.

4 Numerical Results

Performance of the MMPP-based GLRT applied is measured using an intrusion detection evaluation data set [6] developed in 1999 by the Massachusetts Institute of Technology, Lincoln Laboratories under the sponsorship of the Defense Advanced Research Projects Agency (DARPA) and the Air Force Research Laboratory. Henceforth, this data set is referred to as the DARPA data set. This data set consists of five weeks of simulated network traffic, generated using statistics obtained from a real network located on a United States Air Force base. Of the five weeks, we use data from weeks 1–3: weeks 1 and 3 have no attacks, week 2 contains labeled attacks and weeks 4 and 5 contain unlabeled data. We used the portion of the database corresponding to packets resulting from communications between external and internal computers. EM estimation and GLRT classification algorithms were implemented in Matlab on a machine with a 2.93 GHz Intel Xeon X7350 processor.

4.1 Estimation of ϕ_0

Let y^n denote the training signal used to estimate ϕ_0 consisting of all week 1 packet inter-arrival times, with $n = 7293600$. We conducted numerical experiments assuming three parametric forms for $p(y^n; \phi)$: an MMPP, a mixture of exponentials, and a Poisson process, i.e., an MMPP with $r = 1$.

4.1.1 Poisson Process

The Poisson process is parameterized only by the intensity λ, thus $\phi = \lambda$. The pdf is given by $p(y^n; \phi) = \prod_{t=1}^{n} \lambda \exp(-\lambda y_t)$. Let $\tilde{\lambda}$ denote the ML estimate of λ given by $\tilde{\lambda} = n / \sum_{t=1}^{n} y_t = 18.418$. The estimation of $\tilde{\lambda}$ took 0.05 seconds using the computing configuration specified above.

4.1.2 Mixture of Exponentials

A mixture of r exponentials has pdf

$$p(y_t; \phi) = \sum_{i=1}^{r} \alpha(i)\lambda(i)e^{-\lambda(i)y_t},$$

where $\{\alpha(i)\}$ are the mixture weights and $\{\lambda(i)\}$ are the exponential rates. Thus $\phi = \{\lambda(1), \ldots \lambda(r), \alpha(1), \ldots, \alpha(r)\}$ is the parameter of the mixture model. An EM algorithm to estimate ϕ is given by

$$\hat{\lambda}^{k+1}(i) = \frac{\sum_t \xi_t(i; \hat{\phi}^k)}{\sum_t \xi_t(i; \hat{\phi}^k) y_t}, \quad \hat{\alpha}^{k+1}(i) = \frac{\sum_t \xi_t(i; \hat{\phi}^k)}{n} \qquad (7)$$

where conditional probabilities $\xi_t(i; \hat{\phi}^k)$ are calculated using

$$\xi_t(i; \hat{\phi}^k) = \frac{\hat{\alpha}^k(i)\hat{\lambda}^k(i) \exp(-\hat{\lambda}^k(i) y_t)}{p(y_t; \hat{\phi}^k)}$$

We chose the number of states $r = 4$ to allow for diverse traffic patterns while keeping computational overhead to a minimum. Additional parameters $\hat{\lambda}^0$ and $\hat{\alpha}^0(i)$ were chosen as $\hat{\lambda}^0 = (1000, 100, 10, 1)^T$ and $\hat{\alpha}^0(i) = 1/4$ for $i = 1, \ldots, 4$ in order to capture behaviors of different orders of magnitude. Because parameters are estimated using an EM algorithm, values of parameters are subject to local extrema.

Using $\epsilon = 10^{-4}$ in Equation (6), the EM algorithm converged in $k = 19$ iterations with $\log p(y^n; \hat{\phi}^k)/n = 3.406$. The resulting estimates were

$$\hat{\lambda}^k = \begin{pmatrix} 1023.740 \\ 79.725 \\ 20.163 \\ 2.069 \end{pmatrix}, \hat{\alpha}^k = \begin{pmatrix} 0.435 \\ 0.409 \\ 0.062 \\ 0.095 \end{pmatrix} \quad (8)$$

Each iteration of the EM algorithm took approximately 6.4 seconds on the computing configuration specified above.

4.1.3 MMPP

The MMPP EM algorithm used $r = 4$, the same number of states as the exponential mixture model. Estimates for the parameters $\{Q, \Lambda\}$ must be initialized for MMPP training. The diagonal elements of the initial estimate $\hat{\Lambda}^0$ were the final estimates of the mixture of exponentials given in Equation (8): $\hat{\Lambda}^0 = \text{diag}(\hat{\lambda}^k)$. Let A denote the $r \times r$ empirical transition matrix of the exponential mixture states, where the state S^{y_t} during y_t is considered to be the exponential mixture $i = 1 \ldots n$ with largest conditional probability. The initial estimate $\hat{Q}^0$ is given by $\hat{Q}^0 = \log(A)\tilde{\lambda}$. If A has negative eigenvalues, $\hat{Q}^0$ will not be a valid generator matrix. In this case, the rows of $\hat{Q}^0$ can be scaled to produce a valid generator matrix.

Using $\epsilon = 10^{-4}$ in Equation (6), Rydén's EM algorithm converged in $k = 59$ iterations with $\log p(y^n; \hat{\phi}^k)/n = 3.457$. The resulting estimates were

$$\hat{\Lambda}^k = \text{diag}(556.587, 39.232, 0.030, 0.828),$$

$$\hat{Q}^k = \begin{pmatrix} -298.766 & 23.529 & 275.238 & 1.94 \cdot 10^{-7} \\ 17.974 & -40.703 & 7.447 & 15.282 \\ 98.148 & 53.904 & -152.052 & 6.56 \cdot 10^{-6} \\ 1.286 & 0.127 & 0.159 & -1.572 \end{pmatrix}$$

Each iteration of the EM algorithm took approximately 42.7 minutes using the computing configuration specified above.

4.2 Implementation and Performance of GLRT

Assume now that y^n denotes a test sequence that we wish to classify using the GLRT as arising from H_0 or H_1. The GLRT is implemented using the estimates of ϕ_0 given in the previous section. The denominator of Equation (1) is calculated using the ML estimate of ϕ where $p(y^n; \phi)$ is assumed to be a Poisson process. This assumption is made as estimation is simplified considerably compared to estimation when an MMPP is assumed. Furthermore, n is generally too small to produce reliable MMPP estimates on test intervals. With this assumption, the GLRT test statistic comprised of Equations (1)–(2) is given by

$$\log \delta(y^n; \phi_0) = \log p(y^n; \phi_0) - n(\log \tilde{\lambda} - 1).$$

Performance of the models was evaluated on test data containing so-called SYN flood attacks obtained from week 2 of the DARPA data set. The target computers of such attacks are inundated with network packets requesting that the target establishes a connection with a remote machine. The target can become overwhelmed when such requests are left unresolved. There are two SYN flood attacks, each of which are approximately 206 seconds long. This data was segmented into 16, 30-second intervals. From week 3, a week with no attacks, we selected 12, 3-minute intervals of bursty traffic, for a total of 72, 30-second intervals of test data free of attacks. These 88, 30-second intervals each constitute a y^n used in our experiments.

The empirical ROC curves are shown in Figure 1 by plotting the relative frequencies of detections and false alarms for varying thresholds. The curves for the Poisson process, mixture of exponentials, and MMPP is shown as a dashed, dotted-dashed, and solid line, respectively. The curve representing completely random guesses is shown as a dotted line. On our computer configuration, the MMPP classifier operated at speeds of approximately 50 times real time, i.e. the 30 second test signals were classified in just under one second. The mixture of exponential classifier operated at speeds approximately 2 orders of magnitude faster.

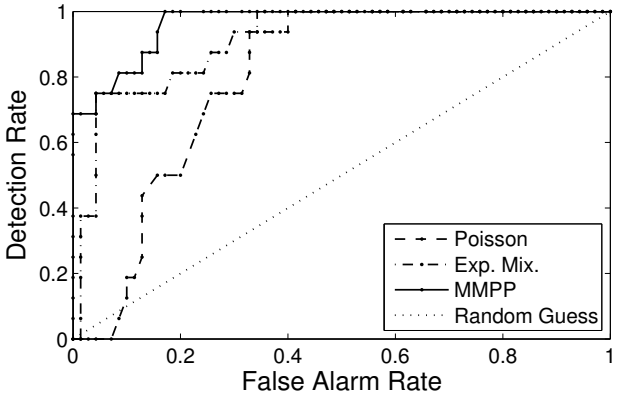

Figure 1: Empirical ROC curves obtained using DARPA data set. The relative frequencies of false alarms and attack detections are plotted for each detection method: Poisson model, exponential mixture model, and MMPP. The dotted line indicating completely random guesses is also shown.

Comparing the ROC curves in Figure 1, we can see that the MMPP achieves lift over the exponential mixture model, which achieves lift over the Poisson model.

Let f_D and f_{FA} denote the relative rates of attack detection and false alarms, respectively. Of particular interest is the region of the low false alarm rate near $f_{FA} = 0$. The MMPP-based GLRT is able to detect 11 of 16 attack segments before suffering a single false alarms. Both the Poisson process and the exponential mixture produce at least one false alarm without successfully detecting any attacks. In this region, the MMPP produces a lower false alarm rate than the other two methods. At full detection ($f_D = 1$), the MMPP-based GLRT suffers significantly fewer false alarms. When $f_D = 1$, out of the 72 segments tested, the MMPP based GLRT produces 12 false alarms. At the same detection rate, the GLRT assuming Poisson and mixture of exponentials, suffer 28 and 24 false alarms, respectively. At peak performance, the MMPP-based GLRT detects 81.25% of attacks with a false alarm rate of 8.57%.

5 Conclusions and Comments

The ROC curve shown in Figure 1 shows that the each model in the GLRT achieves lift over those models that are less sophisticated, indicating that detection performance of each model in the GLRT increases with model sophistication. Using an MMPP yields the highest performance, suggesting that its assumption of Markovian rates is representative of real traffic. The mixture of exponentials is a less elaborate model, assuming iid observations, but requires substantially less computation. The Poisson processes is the simplest model, assuming iid observations and a single traffic rate, but it has very low computational requirements.

The low computational requirements of the GLRTs using a Poisson process and a mixture of exponentials may allow them to be used advantageously in a multi-tiered approach. In this approach, the Poisson-based GLRT is applied first. If H_1 is chosen, the GLRT with a mixture of exponentials is applied. If this classifier also chooses H_1, the MMPP-based GLRT is applied. H_0 is chosen as the final result if any of the individual classifiers choose it, otherwise H_1 is chosen. The thresholds for the tests may need to be carefully chosen. The multi-tiered approach may be particularly applicable for networks which carry significantly more traffic than the network considered here.

References

[1] E. Conte, A. D. Maio, and G. Ricci, "GLRT-based adaptive detection algorithms for range-spread targets," *IEEE Transactions on Signal Processing*, vol. 49, no. 7, pp. 1336–1348, 2001.

[2] L. Deng and J. W. Mark, "Parameter estimation for Markov modulated Poisson processes via the EM algorithm with time discretization," *Telecommunication Systems*, vol. 1, pp. 321–338, 1993.

[3] W. Fischer and K. M. Hellstern, "The Markov-modulated Poisson process (MMPP) cookbook," *Performance Evaluation*, vol. 18, pp. 149–171, 1992.

[4] R. Goel, A. Sardana, and R. Joshi, "Parallel misuse and anomaly detection model," *International Journal of Network Security*, vol. 14, no. 4, pp. 211–222, 2012.

[5] M. Gutman, "Asymptotically optimal classification for multiple tests with empirically observed statistics," *IEEE Transactions on Information Theory*, vol. 35, pp. 401–408, 1989.

[6] J. W. Haines, R. P. Lippmann, D. J. Fried, M. A. Zissman, and E. Tran, "1999 DARPA intrusion detection evaluation: design and procedures," *MIT Lincoln Laboratory Technical Report*, no. 1062, 2001.

[7] H. Heffes, "A class of data traffic processes – covariance function characterization and related queuing results," *Bell System Technical Journal*, vol. 59, no. 6, 1980.

[8] H. Heffes and D. M. Lucantoni, "A Markov modulated characterization of packetized voice and data traffic and related statistical multiplexer performance," *IEEE Journal on Selected Areas in Communications*, vol. 4, no. 6, pp. 856–868, 1986.

[9] D. P. Heyman and D. Lucantoni, "Modeling multiple IP traffic streams with rate limits," *IEEE/ACM Transactions on Networking*, vol. 11, pp. 948–958, 2003.

[10] W. Hoeffding, "Asymptotically optimal tests for multinomial distributions," *The Annals of Mathematical Statistics*, vol. 36, pp. 369–401, 1965.

[11] A. Ihler, J. Hutchins, and P. Smyth, "Learning to detect events with Markov-modulated Poisson processes," *ACM Transactions on Knowledge Discovery from Data*, vol. 1, no. 3, 2007.

[12] Y. Kim, J. Y. Jo, and K. K. Suh, "Baseline profile stability for network anomaly detection," *International Journal of Network Security*, vol. 6, no. 1, pp. 60–66, 2008.

[13] S. Kraut and L. L. Scharf, "The CFAR adaptive subspace detector is a scale-invariant GLRT," *IEEE Transactions on Signal Processing*, vol. 47, pp. 2538–2541, 1999.

[14] E. L. Lehmann, *Testing Statistical Hypotheses, 2nd ed.* Chapman and Hall, 1994.

[15] H. Luo, B. Fang, X. Yun, and Z. Wu, "An effective anomaly detection method in SMTP traffic," *International Journal of Network Security*, vol. 6, no. 3, pp. 321–330, 2008.

[16] N. Merhav, "The estimation of the model order in exponential families," *IEEE Transactions on Information Theory*, vol. 35, pp. 1109–1114, 1989.

[17] L. Muscariello, M. Mellia, M. Meo, M. A. Marsan, and R. L. Cigno, "An MMPP-based hierarchical model of internet traffic," *2004 IEEE International Conference on Communications*, vol. 4, pp. 2143–2147, 2004.

[18] A. Pawling, T. Schoenharl, P. Yan, and G. Madey, "WIPER: An emergency response system," in *Proceedings of 5th International ISCRAM Conference*, Washington, DC, 2008.
[19] W. J. J. Roberts, Y. Ephraim, and E. Dieguez, "On Rydén's EM algorithm for estimating MMPPs," *IEEE Signal Processing Letters*, vol. 13, pp. 373–376, 2006.
[20] W. J. J. Roberts, Y. Ephraim, and H. W. Sabrin, "Speaker classification using composite hypothesis testing and list decoding," *IEEE Transactions on Speech and Audio Processing*, vol. 13, pp. 211–219, 2005.
[21] T. Rydén, "An EM algorithm for estimation in Markov-modulated Poisson processes," *Computational Statistics & Data Analysis*, vol. 21, pp. 431–447, 1996.
[22] S. L. Scott. *Bayesian methods and extensions for the two state Markov modulated Poisson process*. PhD thesis, Department of Statistics, Harvard University, Cambridge, MA, 1998.
[23] S. L. Scott and P. Smyth, "The Markov modulated Poisson process and Markov Poisson cascade with applications to web traffic data," *Bayesian Statistics*, vol. 7, pp. 671–680, 2003.
[24] X. Tang, N. Manikopoulos, and S. G. Ziavras, "Generalized anomaly detection model for Windows-based malicious program behavior," *International Journal of Network Security*, vol. 7, no. 3, pp. 428–435, 2008.
[25] Z. Tian and G. B. Giannakis, "A GLRT approach to data-aided timing acquisition in UWB radios – part i: algorithms," *IEEE Transactions on Wireless Communications*, vol. 4, pp. 2956–2967, 2005.
[26] H. L. Van Trees, *Detection, estimation, and modulation theory, part I.* New York, NY: Wiley Inter-Science, 1968.
[27] C. J. Willy, W. J. J. Roberts, T. A. Mazzuchi, and S. Sarkani, "Recursions for the MMPP score vector and observed information matrix," *Stochastic Models*, vol. 26, pp. 649–665, 2010.
[28] O. Zeitouni, J. Ziz, and N. Merhav, "When is the generalized likelihood ratio test optimal?," *IEEE Transactions on Information Theory*, vol. 38, pp. 1597–1602, 1992.
[29] Z. Zhang, H. Shen, and Y. Sang, "An observation-centric analysis on the modeling of anomaly-based intrusion detection," *International Journal of Network Security*, vol. 4, no. 3, pp. 292–305, 2007.
[30] J. Ziv, "On classification with empirically observed statistics and universal data compression," *IEEE Transactions on Information Theory*, vol. 34, pp. 278–286, 1988.

Chris Scheper Chris Scheper attended Purdue University in West Lafayette, Indiana, USA and received degrees in Mathematics (hons.) and Applied Physics in 2005. He received the Ph.D. degree in Applied Mathematics from Cornell University in Ithaca, New York, USA in 2011. Since 2011, he has been employed at Opera Solutions, LLC, and Opera Solutions Government Services in San Diego, CA, USA.

William J. J. Roberts William J.J. Roberts attended the University of Adelaide, Adelaide, Australia and received, in 1989 and 1990 respectively, honors degrees in Electrical and Electronic Engineering, B.E (hons.), and Science, B.Sc. (hons.), with a double major in Mathematics and Computing. He received the Ph.D. degree in Information Technology from George Mason University, Fairfax, Virginia, USA in 1996. From 1990-2000 he was employed at the Defence Science Technology Organisation, Salisbury, Australia. From 2000–2011, he was employed at Atlantic Coast Technologies, Inc., Silver Spring, MD, USA. Since 2011 he has been employed at Opera Solutions, Jersey City, New Jersey USA.

Cryptanalysis of an ID-based Authenticated Dynamic Group Key Agreement with Optimal Round

Qingfeng Cheng[1,3] and Chunming Tang[2]
(Corresponding author: Qingfeng Cheng)
Department of Language Engineering, Luoyang University of Foreign Languages[1]
Luoyang 471003, P.R. China
(Email: qingfengc2008@sina.com)
School of Mathematics and Information Science, Guangzhou University[2]
Guangzhou 510006, P.R. China
State Key Laboratory of Information Security, Institute of Software,Chinese Academy of Sciences[3]
Beijing 100093, P.R. China
(Received Apr. 28, 2013; revised and accepted Mar. 15 & Apr. 25, 2014)

Abstract

Recently, Teng, Wu and Tang proposed a new ID-based authenticated dynamic group key agreement (DGKA) protocol. They claimed that leaving users cannot calculate subsequent group session keys and joining users cannot calculate previous group session keys. In this paper, we will show that Teng et al.'s protocol cannot provide forward confidentiality or backward confidentiality.

Keywords: Backward confidentiality, bilinear pairing, dynamic group key agreement, forward confidentiality

1 Introduction

Ingemarsson et al. [4] first introduced the concept of group key agreement (GKA). Afterward, many group key agreement protocols have been proposed [1, 2, 3, 5]. In particular, some of them are designed for dynamic groups, which are called dynamic group key agreement (DGKA) protocols. Secure DGKA protocols must provide the fundamental security requirements for general GKA protocols, and also should encompass the following two requirements [6, 8]:

- Forward confidentiality: While a group user leaves from the current group, he should not be able to calculate the new session key.
- Backward confidentiality: While a new user joins into the current group, he should not be able to calculate the previous session key.

Recently, Teng, Wu and Tang [7] proposed a new ID-based authenticated DGKA protocol, called Teng-Wu-Tang protocol. They proved the Teng-Wu-Tang protocol's security in the random oracle model. In addition, they also claimed that the leaving users cannot obtain information about subsequent new group session keys and joining users cannot obtain information about previous group session keys. In this paper, we will demonstrate that the Teng-Wu-Tang protocol is not secure. Though the Teng-Wu-Tang protocol's join algorithm only requires one round communication and its leave algorithm does not require exchange message, the Teng-Wu-Tang protocol cannot provide forward confidentiality or backward confidentiality. It means that the Teng-Wu-Tang protocol is infeasible for real-life implementation.

2 Review of the Teng-Wu-Tang Protocol

In this section, we briefly review the Teng-Wu-Tang protocol, which is composed of the following three stages as well as the join algorithm and the leave algorithm. For more details, refer to [7].

2.1 System Initialization Stage

Let q be a large prime, G_1 and G_2 be two groups with the same order of q. P is a generator of G_1 and Q is randomly chosen from G_1. $\hat{e} : G_1 \times G_1 \to G_2$ is a bilinear pairing and $H : \{0,1\}^* \to G_1^*$ is a hash function. Key generation center (KGC) randomly chooses the master private key $s \in Z_q^*$ and computes $P_{pub} = sP$ as the master public key. The system parameters are $\{q, G_1, G_2, P, Q, \hat{e}, H, P_{pub}\}$.

2.2 Key Extract Stage

This phase is run by the KGC for each user with an identity $ID_i \in \{0,1\}^*$. The KGC first computes $Q_i = H(ID_i)$, and then computes the user's private key $S_i = sQ_i$.

2.3 Key Agreement Stage

Let $\{U_1, \ldots, U_n\}$ be the initial group of n users. The key agreement stage is described below:

1) Each user $U_i (1 \le i \le n)$ define the $(n-1) \times n$ matrix

$$A = \begin{pmatrix} a_2 \\ a_3 \\ \cdot \\ \cdot \\ \cdot \\ a_n \end{pmatrix} = \begin{pmatrix} 1 & 1 & 0 & \cdot & \cdot & \cdot & 0 \\ 1 & 0 & 1 & \cdot & \cdot & \cdot & 0 \\ \cdot & \cdot & \cdot & \cdot & & & \\ \cdot & \cdot & \cdot & \cdot & \cdot & & \\ 1 & \cdot & \cdot & \cdot & 1 & 0 & \\ 1 & 0 & 0 & \cdot & \cdot & 0 & 1 \end{pmatrix}$$

2) Each user $U_i (1 \le i \le n)$ randomly chooses $r_i \in Z_q^*$ and computes $P_i = r_i P, V_{i_j} = r_i Q'_j (1 \le j \le n, j \ne i)$, where $Q'_j = Q + Q_j$. Then user U_i broadcasts P_i, V_{i_j}.

3) Each user $U_i (1 < i < n)$ sets two vectors $a_{i_1} = (0, ..., 0, 1_i, 0, 0, ..., 0)$, which denotes ith element is 1, and $a'_{i_1} = (0, ..., 0, 1_{i+1}, 0, 0, ..., 0)$ with n elements, which denotes $i+1$th element is 1, and then defines two $n \times n$ matrixes A_i and A'_i as follows:

$$A_i = \begin{pmatrix} a_{i_1} \\ a_2 \\ \vdots \\ a_n \end{pmatrix}, \qquad A'_i = \begin{pmatrix} a'_{i_1} \\ a_2 \\ \vdots \\ a_n \end{pmatrix}$$

User U_1 sets a vector $a'_{1_1} = (0, 1, 0, ..., 0)$ with n elements. User U_n also sets a vector $a_{n_1} = (0, 0, ..., 1)$ with n elements. Then U_1 and U_n define two matrixes A'_1 and A_n respectively as follows:

$$A'_1 = \begin{pmatrix} a_{1'_1} \\ a_2 \\ \vdots \\ a_n \end{pmatrix}, \qquad A_n = \begin{pmatrix} a_{n_1} \\ a_2 \\ \vdots \\ a_n \end{pmatrix}$$

Two matrixes A_i and A'_i are nonsingular due to $|A_i| \ne 0$ and $|A'_i| \ne 0$, where $|\cdot|$ denotes the determinant of a matrix. Let $(x_1, x_2, \ldots, x_n)$ be the solution of $X \times A_i = (1, 1, ..., 1)$ and $(x'_1, x'_2, \ldots, x'_n)$ be the solution of $X \times A'_i = (1, 1, ..., 1)$.

Further, U_i defines the $(n-1) \times (n-1)$ matrix M_i as follows:

$$M_i = \begin{pmatrix} V_{1_2} & \cdots & V_{(i-1)_1} & V_{(i+1)_1} & \cdots & V_{n_1} \\ V_{1_3} & \cdots & V_{(i-1)_2} & V_{(i+1)_2} & \cdots & V_{n_2} \\ & \vdots & & \vdots & & \\ V_{1_n} & \cdots & V_{(i-1)_n} & V_{(i+1)_n} & \cdots & V_{n_{(n-1)}} \end{pmatrix}$$

It means that messages V_{c_d} from the cth column is received from user c, when $c < i$, and also messages $V_{(c+1)_d}$ from the cth column is received from user $c+1$, when $c \ge i$.

Next, U_i sets two matrixes $M_{i,1}$ and $M_{i,2}$, which are composed of the first $i-1$ columns of the matrix M_i and the other columns of the matrix M_i respectively.

$$M_{i,1} = \begin{pmatrix} r_1(Q+Q_2) & \cdots & r_{i-1}(Q+Q_1) \\ r_1(Q+Q_3) & \cdots & r_{i-1}(Q+Q_2) \\ \vdots & & \vdots \\ r_1(Q+Q_{i-2}) & \cdots & r_{i-1}(Q+Q_{i-2}) \\ r_1(Q+Q_{i-1}) & \cdots & r_{i-1}(Q+Q_i) \\ \vdots & & \vdots \\ r_1(Q+Q_n) & \cdots & r_{i-1}(Q+Q_n) \end{pmatrix}$$

$$M_{i,2} = \begin{pmatrix} r_{i+1}(Q+Q_1) & \cdots & r_n(Q+Q_1) \\ r_{i+1}(Q+Q_2) & \cdots & r_n(Q+Q_2) \\ \vdots & & \vdots \\ r_{i+1}(Q+Q_i) & \cdots & r_n(Q+Q_i) \\ r_{i+1}(Q+Q_{i+2}) & \cdots & r_n(Q+Q_{i+1}) \\ \vdots & & \vdots \\ r_{i+1}(Q+Q_n) & \cdots & r_n(Q+Q_{n-1}) \end{pmatrix}$$

With $(x_1, x_2, \cdots, x_n)$ and $(x'_1, x'_2, \cdots, x'_n)$, U_i computes

$$\hat{Q}_{i,1} = (x_2, x_3, \cdots, x_n) M_{i,1} \begin{pmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{pmatrix}_{n-1}$$

$$\hat{Q}_{i,2} = (x'_2, x'_3, \cdots, x'_n) M_{i,2} \begin{pmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{pmatrix}_{n-1}$$

Finally, $U_i (1 < i < n)$ computes the group session key

$$sk = sk_{i,1} \cdot sk_{i,2} \cdot \hat{e}(P_{pub}, r_i \overline{Q_i}) = \prod_{i=1}^{i=n} \hat{e}(P_{pub}, \overline{Q_i})^{r_i},$$

where

$$sk_{i,1} = \hat{e}(\sum_{j=1}^{i-1} P_j, x_1 S_i) \cdot \hat{e}(P_{pub}, \hat{Q}_{i,1}),$$

$$sk_{i,2} = \hat{e}(\sum_{j=i+1}^{n} P_j, x'_1 S_i) \cdot \hat{e}(P_{pub}, \hat{Q}_{i,2})$$

and $\overline{Q_i} = Q + Q_1 + Q_2 + \cdots + Q_{i-1} + Q_{i+1} + \cdots + Q_n$ $(1 \le i \le n)$.

User U_1 computes the group session key $sk = sk_{1,2} \cdot \hat{e}(P_{pub}, r_1 \overline{Q_1}) = \prod_{i=1}^{i=n} \hat{e}(P_{pub}, \overline{Q_i})^{r_i}$ and user U_n computes the group session key $sk = sk_{n,1} \cdot \hat{e}(P_{pub}, r_n \overline{Q_n}) = \prod_{i=1}^{i=n} \hat{e}(P_{pub}, \overline{Q_i})^{r_i}$.

2.4 Join Algorithm

Let $\{U_1, \ldots, U_n\}$ be the current group and $\{U_{n+1}, \ldots, U_{n+m}\}$ be the set of joining users. For generating the new group session key, each user $U_i (1 \le i \le n+m)$ first defines a new $(n+m-1) \times (n+m)$ matrix A as follows:

$$A = \begin{pmatrix} a_2 \\ a_3 \\ \cdot \\ \cdot \\ \cdot \\ a_m \end{pmatrix} = \begin{pmatrix} 1 & 1 & 0 & \cdot & \cdot & \cdot & 0 \\ 1 & 0 & 1 & \cdot & \cdot & \cdot & 0 \\ \cdot & \cdot & \cdot & \cdot & & & \\ \cdot & \cdot & \cdot & \cdot & \cdot & & \\ 1 & \cdot & \cdot & \cdot & 1 & 0 & \\ 1 & 0 & 0 & \cdot & \cdot & 0 & 1 \end{pmatrix}$$

Then each user $U_i (1 \le i \le n)$ computes $r_i(Q + Q_{n+j})(1 \le j \le m)$, where r_i is chosen in the key agreement stage. Then user $U_i (1 \le i \le n)$ broadcasts $r_iP, r_i(Q + Q_{j'})(1 \le j' \le n+m, j' \ne i)$, where r_iP and $r_i(Q + Q_{j'})(1 \le j' \le n, j' \ne i)$ are computed in the key agreement stage. At the same time, user $U_{n+j} (1 \le j \le m)$ randomly chooses $r_{n+j} \in Z_q^*$, computes and broadcasts $P_{n+j} = r_{n+j}P, V_{(n+j)_{j'}} = r_{n+j}(Q + Q_{j'})(1 \le j \le m, 1 \le j' \le n+m, j' \ne n+j)$, where r_{n+j} is kept secretly.

Finally, each user $U_i (1 \le i \le n+m)$ computes the new group session key as $sk = \prod_{i=1}^{i=n+m} \hat{e}(P_{pub}, \overline{Q_i})^{r_i}$, where $\overline{Q_i} = Q + Q_1 + Q_2 + \cdots + Q_{i-1} + Q_{i+1} + \cdots + Q_{n+m}$.

2.5 Leave Algorithm

Let $\{U_{m+1}, \ldots, U_n\}$ be the set of leaving users and $\{U_1, \ldots, U_m\}$ be the current group. For generating the new group session key, each user $U_i (1 \le i \le m)$ first defines a new $(m-1) \times m$ matrix A as follows:

$$A = \begin{pmatrix} a_2 \\ a_3 \\ \cdot \\ \cdot \\ \cdot \\ a_m \end{pmatrix} = \begin{pmatrix} 1 & 1 & 0 & \cdot & \cdot & \cdot & 0 \\ 1 & 0 & 1 & \cdot & \cdot & \cdot & 0 \\ \cdot & \cdot & \cdot & \cdot & & & \\ \cdot & \cdot & \cdot & \cdot & \cdot & & \\ 1 & \cdot & \cdot & \cdot & 1 & 0 & \\ 1 & 0 & 0 & \cdot & \cdot & 0 & 1 \end{pmatrix}$$

Then each user $U_i (1 \le i \le m)$ defines the new $(m-1) \times (m-1)$ matrix M_i', which includes the first $m-1$ rows and the first $m-1$ columns of matrix M_i.

Finally, each user $U_i (1 \le i \le m)$ computes the new group session key as $sk = \prod_{i=1}^{i=m} \hat{e}(P_{pub}, \overline{Q_i})^{r_i}$, where $\overline{Q_i} = Q + Q_1 + Q_2 + \cdots + Q_{i-1} + Q_{i+1} + \cdots + Q_m$.

3 Cryptanalysis of the Teng-Wu-Tang Protocol

In this section, we show that the Teng-Wu-Tang protocol is not secure. Here, we only consider the simplest case, i.e. a joining user and a leaving user. In the join algorithm, a new joining user can use his private key and ephemeral key to recover the accepted group session key generated by the former group users before he joined the group. In the leave algorithm, a leaving user can use his private key and ephemeral key to compute the new group session key generated by the new group users after he left the group.

3.1 Attack on the Backward Confidentiality

Let $\{U_1, \ldots, U_n\}$ be the set of the current group users and U_{n+1} be the new joining user. From the key agreement stage, we know that $U_1, U_2 \ldots, U_{n-1}$ and U_n have shared the group session key $sk = \prod_{i=1}^{i=n} \hat{e}(P_{pub}, \overline{Q_i})^{r_i}$, where $\overline{Q_i} = Q + Q_1 + Q_2 + \cdots + Q_{i-1} + Q_{i+1} + \cdots + Q_n$. In order to keep the previous encrypted messages secretly, $U_1, U_2 \ldots, U_n$ and U_{n+1} must use the join algorithm to generate a new shared group session key $sk' = \prod_{i=1}^{i=n+1} \hat{e}(P_{pub}, \overline{Q_i'})^{r_i}$, where $\overline{Q_i'} = Q + Q_1 + Q_2 + \cdots + Q_{i-1} + Q_{i+1} + \cdots + Q_n + Q_{n+1}$.

If U_{n+1} is a malicious user, he can use his private key S_{n+1}, ephemeral key r_{n+1} and the new group session key sk' to recover the previous group session key.

Since

$$\begin{aligned} sk' &= \prod_{i=1}^{i=n+1} \hat{e}(P_{pub}, \overline{Q_i'})^{r_i} \\ &= \left[\prod_{i=1}^{i=n} \hat{e}(P_{pub}, \overline{Q_i'})^{r_i}\right] \hat{e}(P_{pub}, \overline{Q_{n+1}'})^{r_{n+1}} \\ &= \left[\prod_{i=1}^{i=n} \hat{e}(P_{pub}, \overline{Q_i} + Q_{n+1})^{r_i}\right] \hat{e}(P_{pub}, \overline{Q_{n+1}'})^{r_{n+1}} \\ &= \left[\prod_{i=1}^{i=n} \hat{e}(P_{pub}, \overline{Q_i})^{r_i}\right] \left[\prod_{i=1}^{i=n} \hat{e}(P_{pub}, Q_{n+1})^{r_i}\right] \\ &\qquad \hat{e}(P_{pub}, \overline{Q_{n+1}'})^{r_{n+1}} \\ &= sk \left[\prod_{i=1}^{i=n} \hat{e}(P_{pub}, Q_{n+1})^{r_i}\right] \hat{e}(P_{pub}, \overline{Q_{n+1}'})^{r_{n+1}} \\ &= sk \left[\prod_{i=1}^{i=n} \hat{e}(P, sQ_{n+1})^{r_i}\right] \hat{e}(P_{pub}, \overline{Q_{n+1}'})^{r_{n+1}} \\ &= sk \left[\prod_{i=1}^{i=n} \hat{e}(r_iP, sQ_{n+1})\right] \hat{e}(P_{pub}, \overline{Q_{n+1}'})^{r_{n+1}} \\ &= sk \left[\prod_{i=1}^{i=n} \hat{e}(P_i, S_{n+1})\right] \hat{e}(P_{pub}, \overline{Q_{n+1}'})^{r_{n+1}}, \end{aligned}$$

the malicious user U_{n+1} can compute $\prod_{i=1}^{i=n} \hat{e}(P_i, S_{n+1})$ and $\hat{e}(P_{pub}, \overline{Q_{n+1}'})^{r_{n+1}}$ with his private key S_{n+1} and ephemeral key r_{n+1}, where $P_i = r_iP (1 \le i \le n)$ is a public message and $\overline{Q_{n+1}'} = Q + Q_1 + Q_2 + \cdots + Q_n$.

Then the malicious user U_{n+1} can compute the previous group session key as follows:

$$sk = \frac{sk'}{\left[\prod_{i=1}^{i=n} \hat{e}(P_i, S_{n+1})\right] \hat{e}(P_{pub}, \overline{Q_{n+1}'})^{r_{n+1}}}$$

Clearly, the new joining user U_{n+1} has successfully computed the previous group session key generated be-

fore he joined the current group. Therefore, the Teng-Wu-Tang protocol cannot provide backward confidentiality.

3.2 Attack on the Forward Confidentiality

Let $\{U_1, \ldots, U_n\}$ be the set of the current group users and U_n be the leaving user. From the key agreement stage, we know that $U_1, U_2 \ldots, U_{n-1}$ and U_n have shared the group session key $sk = \prod_{i=1}^{i=n} \hat{e}(P_{pub}, \overline{Q_i})^{r_i}$, where $\overline{Q_i} = Q+Q_1+Q_2+\cdots+Q_{i-1}+Q_{i+1}+\cdots+Q_n$. In order to keep the future encrypted messages secretly, $U_1, U_2 \ldots, U_{n-2}$ and U_{n-1} must use the leave algorithm to generate a new shared group session key $sk'' = \prod_{i=1}^{i=n-1} \hat{e}(P_{pub}, \overline{Q_i''})^{r_i}$, where $\overline{Q_i''} = Q + Q_1 + Q_2 + \cdots + Q_{i-1} + Q_{i+1} + \cdots + Q_{n-2} + Q_{n-1}$.

If U_n is a malicious user, he can write the current group session key sk as follows:

$$
\begin{aligned}
sk &= \prod_{i=1}^{i=n} \hat{e}(P_{pub}, \overline{Q_i})^{r_i} \\
&= \left[\prod_{i=1}^{i=n-1} \hat{e}(P_{pub}, \overline{Q_i})^{r_i}\right] \hat{e}(P_{pub}, \overline{Q_n})^{r_n} \\
&= \left[\prod_{i=1}^{i=n-1} \hat{e}(P_{pub}, \overline{Q_i''} + Q_n)^{r_i}\right] \hat{e}(P_{pub}, \overline{Q_n})^{r_n} \\
&= \left[\prod_{i=1}^{i=n-1} \left[\hat{e}(P_{pub}, \overline{Q_i''})^{r_i} \hat{e}(P_{pub}, Q_n)^{r_i}\right]\right] \hat{e}(P_{pub}, \overline{Q_n})^{r_n} \\
&= \left[\prod_{i=1}^{i=n-1} \hat{e}(P_{pub}, \overline{Q_i''})^{r_i}\right] \left[\prod_{i=1}^{i=n-1} \hat{e}(P_{pub}, Q_n)^{r_i}\right] \\
&\qquad \hat{e}(P_{pub}, \overline{Q_n})^{r_n} \\
&= sk'' \left[\prod_{i=1}^{i=n-1} \hat{e}(P_{pub}, Q_n)^{r_i}\right] \hat{e}(P_{pub}, \overline{Q_n})^{r_n} \\
&= sk'' \left[\prod_{i=1}^{i=n-1} \hat{e}(r_i P_{pub}, Q_n)\right] \hat{e}(P_{pub}, \overline{Q_n})^{r_n} \\
&= sk'' \left[\prod_{i=1}^{i=n-1} \hat{e}(r_i sP, Q_n)\right] \hat{e}(P_{pub}, \overline{Q_n})^{r_n} \\
&= sk'' \left[\prod_{i=1}^{i=n-1} \hat{e}(r_i P, sQ_n)\right] \hat{e}(P_{pub}, \overline{Q_n})^{r_n} \\
&= sk'' \left[\prod_{i=1}^{i=n-1} \hat{e}(P_i, S_n)\right] \hat{e}(P_{pub}, \overline{Q_n})^{r_n}.
\end{aligned}
$$

Since S_n is the private key of user U_n and r_n is selected by user U_n, he can compute $\prod_{i=1}^{i=n-1} \hat{e}(P_i, S_n)$ and $\hat{e}(P_{pub}, \overline{Q_n})^{r_n}$, where $P_i = r_i P (1 \le i \le n-1)$ is a public message and $\overline{Q_n} = Q + Q_1 + Q_2 + \cdots + Q_{n-1}$. Finally, the malicious user U_n can compute the new group session key as follows:

$$
sk'' = \frac{sk}{\left[\prod_{i=1}^{i=n-1} \hat{e}(P_i, S_n)\right] \hat{e}(P_{pub}, \overline{Q_n})^{r_n}}
$$

Clearly, the leaving user U_n has successfully computed the new group session key generated after he left the current group. Therefore, the Teng-Wu-Tang protocol cannot provide forward confidentiality.

4 Conclusion

In this paper, we have pointed out that the Teng-Wu-Tang protocol fails to provide forward confidentiality and backward confidentiality. It means that a leaving user can calculate the future session key and a joining user can calculate the previous session key. So the Teng-Wu-Tang protocol is not suitable for practical applications.

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China (Grant No. 61202317, 11271003), Province Natural Science Foundation of Guangdong (Grant No. S2012010009950), High Level Talents Project of Guangdong, the Young Key Teacher Foundation of Henan Province's Universities (Grant No. 2012-GGJS-157), and the Foundation of State Key Laboratory of Information Security (No. 2014-11). The authors would like to thank the anonymous referees for their helpful comments.

References

[1] A. H. Ahmed, M. Ali, O. B. Luis, "Authenticated group key agreement protocols for Ad hoc wireless networks," *International Journal of Network Security*, vol. 4, no. 1, pp. 90–98, 2007.

[2] T. Y. Chang, M. S. Hwang, W. P. Yang, "Cryptanalysis of the Tseng-Jan anonymous conference key distribution system without using a one-way hash function," *Information & Security: An International Journal*, vol. 15, no. 1, pp. 110–114, 2004.

[3] S. Hong, "Queue-based group key agreement protocol," *International Journal of Network Security*, vol. 9, no. 2, pp. 135–142, 2009.

[4] I. Ingemarsson, D. Tang, C. Wong, "A conference key distribution system," *IEEE Transactions on Information Theory*, vol. 28, no. 5, pp. 714–720, 1982.

[5] J. Katz, M. Yung, "Scalable protocols for authenticated group key exchange," *Journal of Cryptology*, vol. 20, no. 1, pp. 85–113, 2007.

[6] S. Michael, T. Gene, and W. Michael, "Key agreement in dynamic peer groups," *IEEE Transactions on Parallel and Distributed Systems*, vol. 11, no. 8, pp. 769–780, 2000.

[7] J. K. Teng, C. K. Wu, C. M. Tang, "An ID-based authenticated dynamic group key agreement with optimal round," *Science China Information Sciences*, vol. 55, no. 11, pp. 2542–2554, 2012.
[8] Y. M. Tseng, "A communication-efficient and fault-tolerant conference-key agreement protocol with forward secrecy," *Journal of Systems and Software*, vol. 80, no. 7, pp. 1091–1101, 2007.

Qingfeng Cheng received his B.A. degree in 2000 and M.S. degree in 2004 from National University of Defense Technology, and Ph.D. degree in 2011 from Information Engineering University. He is now an Associate Professor with the Department of Language Engineering, Luoyang University of Foreign Languages. His research interests include cryptography and information security.

Chunming Tang who is born in 1972, and is a professor in Guangzhou University in P.R. China. He received his Bachelors degree in Xiangtan Normal University in 1995, then became an assistant in same university. He received his masters degree from Xiangtan University in 2001, and his Mathematics PhD from Chinese Academy of Sciences in 2004. Since then, he works in Guangzhou University, where he became an associate professor in 2005 and a professor in 2009. His research field is cryptology and cloud computing.

Towards Generating Real-life Datasets for Network Intrusion Detection

Monowar H. Bhuyan[1], Dhruba K. Bhattacharyya[2], and Jugal K. Kalita[3]
(Corresponding author: Monowar H. Bhuyan)
Department of Computer Science and Engineering, Kaziranga University, Jorhat-785006, Assam, India[1]
(Email: monowar.tezu@gmail.com)
Department of Computer Science and Engineering, Tezpur University, Tezpur-784028, Assam, India[2]
(Email: dkb@tezu.ernet.in)
Department of Computer Science, University of Colorado at Colorado Springs, CO 80918, USA[3]
(Email: jkalita@uccs.edu)

(Received February 5, 2015; revised and accepted Apr. 20 & May 9, 2015)

Abstract

With exponential growth in the number of computer applications and the sizes of networks, the potential damage that can be caused by attacks launched over the Internet keeps increasing dramatically. A number of network intrusion detection methods have been developed with respective strengths and weaknesses. The majority of network intrusion detection research and development is still based on simulated datasets due to non-availability of real datasets. A simulated dataset cannot represent a real network intrusion scenario. It is important to generate real and timely datasets to ensure accurate and consistent evaluation of detection methods. In this paper, we propose a systematic approach to generate unbiased full-feature real-life network intrusion datasets to compensate for the crucial shortcomings of existing datasets. We establish the importance of an intrusion dataset in the development and validation process of detection mechanisms, identify a set of requirements for effective dataset generation, and discuss several attack scenarios and their incorporation in generating datasets. We also establish the effectiveness of the generated dataset in the context of several existing datasets.

Keywords: Dataset, intrusion detection, NetFlow, network traffic

1 Introduction

In network intrusion detection, particularly when using anomaly based detection, it is difficult to accurately evaluate, compare and deploy a system that is expected to detect novel attacks due to scarcity of adequate datasets. Before deploying in any real world environment, an anomaly based network intrusion detection system (ANIDS) must be trained, tested and evaluated using real labelled network traffic traces with a intensive set of intrusions or attacks. This is a significant challenge, since not many such datasets are available. Therefore the detection methods and systems are evaluated only with a few publicly available datasets that lack comprehensiveness and completeness [2, 17] or are outdated. For example, Cooperative Association for Internet Data Analysis (CAIDA) Distributed Denial of Service (DDoS) 2007, Lawrence Berkeley National Laboratory (LBNL), and ICSI datasets are heavily anonymized without payload information, decreasing research utility. Researchers also frequently use a single NetFlow based intrusion dataset found at [25, 40] with a limited number of attacks.

1.1 Importance of Datasets

In network traffic anomaly detection, it is always important to test and evaluate detection methods and systems using datasets as network scenarios evolve. We enumerate the following reasons to justify the importance of a dataset.

- *Repeatability of experiments*: Researchers should be able to repeat experiments with the dataset and get similar results, when using the same approach. This is important because the proposed method should cope with the evolving nature of attacks and network scenarios.

- *Validation of new approaches*: New methods and algorithms are being continuously developed to detect network anomalies. It is necessary that every new approach be validated.

- *Comparison of different approaches*: State-of-the-art network anomaly detection methods must not only be validated, but also show improvements over older methods in performance in a quantifiable manner. For example, the

DARPA 1998 dataset [26] is commonly used for performance evaluation of anomaly detection systems [24]. So that one method can be compared against others.

- *Parameters tuning*: To properly obtain the model to classify the normal from malicious traffic, it is necessary to tune model parameters. Network anomaly detection assumes the normality model to identify malicious traffic. For example, Cemerlic et al. [9] and Thomas et al. [44] use the attack-free part of the DARPA 1999 dataset for training to estimate parameter values.

- *Dimensionality or the number of features*: An optimal set of features or attributes should be used to represent normal as well as all possible attack instances.

1.2 Requirements

Although good datasets are necessary for validating and evaluating IDSs, generating such datasets is a time consuming task. A dataset generation approach should meet the following requirements.

- *Real world*: A dataset should be generated by monitoring the daily situation in a realistic way, such as the daily network traffic of an organization.

- *Completeness in labelling*: The labelling of traffic as benign or malicious must be backed by proper evidence for each instance. The aim these days should be to provide labelled datasets at both packet and flow levels for each piece of benign and malicious traffic.

- *Correctness in labelling*: Given a dataset, labelling of each traffic instance must be correct. This means that our knowledge of security events represented by the data has to be certain.

- *Sufficient trace size*: The generated dataset should be unbiased in terms of size in both benign and malicious traffic instances.

- *Concrete feature extraction*: Extraction of an optimal set of concrete features when generating a dataset is important because such features play an important role when validating a detection mechanisms.

- *Diverse attack scenarios*: With the increasing frequency, size, variety and complexity of attacks, intrusion threats have become more complex including the selection of targeted services and applications. When contemplating attack scenarios for dataset generation, it is important to tilt toward a diverse set of multi-step attacks that are recent.

- *Ratio between normal and attack traffic*: Most benchmark datasets are biased because the proportion of normal and attack traffic are not the same. This is because normal traffic is usually much more common than anomalous traffic. However, the evaluation of an intrusion detection method or system using biased datasets may not be fit for real-time deployment in certain situations. Most existing datasets have been created based on the following assumptions.

  – Anomalous traffic is statistically different from normal traffic [13].

  – The majority of network traffic instances is normal [36].

  However, unlike most traditional intrusions, DDoS attacks do not follow these assumptions because they change network traffic rate dynamically and employ multi-stage attacks. A DDoS dataset must reflect this fact.

1.3 Motivation and Contributions

By considering the aforementioned requirements, we propose a systematic approach for generating real-life network intrusion dataset at both packet and flow levels with a view to analyzing, testing and evaluating network intrusion detection methods and systems with a clear focus on anomaly based detectors. The following are the major contributions of this paper.

- We present guidelines for real-life intrusion dataset generation.

- We discuss systematic generation of both normal and attack traffic.

- We extract features from the captured network traffic such as *basic*, *content*-based, *time*-based, and *connection*-based features using a distributed feature extraction framework.

- We generate three categories of real-life intrusion datasets, viz., (i) TUIDS (Tezpur University Intrusion Detection System) intrusion dataset, (ii) TUIDS coordinated scan dataset, and (iii) TUIDS DDoS dataset. These datasets are available for the research community to download for free.

1.4 Organization of the Paper

The remainder of the paper is organized as follows. Section 2 discusses prior datasets and their characteristics. Section 3 is dedicated to the discussion of a systematic approach to generate real-life datasets for intrusion detection with a focus on network anomaly detectors. Finally, Section 4 presents observations and concluding remarks.

2 Existing Datasets

As discussed earlier, datasets play an important role in the testing and validation of network anomaly detection methods or systems. A good quality dataset not only allows us to identify the ability of a method or a system to detect anomalous behavior, but also allows us to provide potential effectiveness when deployed in real operating environments. Several

datasets are publicly available for testing and evaluation of network anomaly detection methods and systems. A taxonomy of network intrusion datasets is shown in Figure 1. We briefly discuss each of them below.

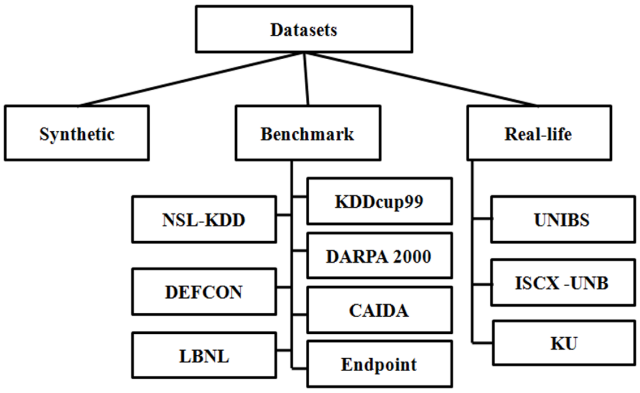

Figure 1: A taxonomy of network intrusion datasets [2]

2.1 Synthetic Datasets

Synthetic datasets are generated to meet specific needs or certain conditions or tests that real data satisfy. Such datasets are useful when designing any prototype system for theoretical analysis so that the design can be refined. A synthetic dataset can be used to test and create many different types of test scenarios. This enables designers to build realistic behavior profiles for normal users and attackers based on the dataset to test a proposed system. This provides initial validation of a specific method or a system; if the results prove to be satisfactory, the developers then continue to evaluate a method or a system in a specific domain real-life data.

2.2 Benchmark Datasets

We discuss seven publicly available benchmark datasets generated using simulated environments in large networks. Different attack scenarios were simulated during the generation of these datasets.

2.2.1 KDDcup99 Dataset

Since 1999, the KDDcup99 dataset [21] has been the most widely used dataset for evaluation of network based anomaly detection methods and systems. This dataset was prepared by Stolfo et al. [41] and is built upon the data captured in the DARPA98 IDS evaluation program. The KDD training dataset consists of approximately $4,900,000$ single connection vectors, each of which contains 41 features and is labelled as either normal or attack of a specific attack type. The test dataset contains about $300,000$ samples with a total 24 training types, with an additional 14 attack types in the test dataset only [14]. The represented attacks are mainly four types: denial of service, remote-to-local, user-to-root, and surveillance or probing.

- *Denial of Service (DoS)*: An attacker attempts to prevent valid users from using a service provided by a system. Examples include SYN flood, smurf and teardrop attacks.
- *Remote to Local (r2l)*: Attackers try to gain entrance to a victim machine without having an account on it. An example is the password guessing attack.
- *User to Root (u2r)*: Attackers have access to a local victim machine and attempt to gain privilege of a superuser. Examples include buffer overflow attacks.
- *Probe*: Attackers attempt to acquire information about the target host. Some examples of probe attacks are portscans and ping-sweep attacks.

Background traffic was simulated and the attacks were all known. The training set, consisting of seven weeks of labelled data, is available to the developers of intrusion detection systems. The testing set also consists of simulated background traffic and known attacks, including some attacks that are not present in the training set. The distribution of normal and attack traffic for this dataset is reported in Table 1. We also identify the services associated with each category of attacks [12, 22] and summarize them in Table 2.

2.2.2 NSL-KDD Dataset

Analysis of the KDD dataset showed that there were two important issues with the dataset, which highly affect the performance of evaluated systems often resulting in poor evaluation of anomaly detection methods [43]. To address these issues, a new dataset known as NSL-KDD [32], consisting of selected records of the complete KDD dataset was introduced. This dataset is also publicly available for researchers[1] and has the following advantages over the original KDD dataset.

- This dataset doesn't contain superfluous and repeated records in the training set, so classifiers or detection methods will not be biased towards more frequent records.
- There are no duplicate records in the test set. Therefore, the performance of learners is not biased by the methods which have better detection rates on frequent records.
- The number of selected records from each difficulty level is inversely proportional to the percentage of records in the original KDD dataset. As a result, the classification rates of various machine learning methods vary in a wider range, which makes it more efficient to have an accurate evaluation of various learning techniques.
- The number of records in the training and testing sets is reasonable, which makes it practical to run experiments on the complete set without the need to randomly select a small portion. Consequently, evaluation results of different research groups are consistent and comparable.

[1] http://www.iscx.ca/NSL-KDD/

Table 1: Distribution of normal and attack traffic instances in KDDCup99 dataset

Dataset	*DoS*		*Probe*		*u2r*		*r2l*		Normal
	Total instances	Attacks	Total instances	Attacks	Total instances	Attacks	Total instances	Attacks	
10% KDD	391458	smurf, neptune, back, teardrop, pod, land	4107	satan, ipsweep, portsweep, nmap	52	buffer_overflow, rootkit, loadmodule, perl	1126	warezclient, guess_passwd, warezmaster, imap, ftp_write, multihop, phf, spy	97277
Corrected KDD	229853		4107		52		1126		97277
Whole KDD	229853		4107		52		1126		97277

Table 2: List of attacks and corresponding services in KDDcup99 dataset

Dataset	*DoS*		*Probe*		*u2r*		*r2l*	
	Attack name	Service(s)	Attack name	Service(s)	Attack name	Service(s)	Attack name	Service(s)
KDD99	apache2	http	ipsweep	icmp	eject	Any user session	dictionary	telnet, rlogin, pop, imap, ftp
	back	http	mscan	many	ffbconfig	Any user session	ftp-write	ftp
	land	N/A	nmap	many	fdformat	Any user session	guest	telnet, rlogin
	mailbomb	smtp	saint	many	loadmodule	Any user session	imap	imap
	SYN flood	Any TCP	satan	many	perl	Any user session	named	dns
	ping of death	icmp			ps	Any user session	named	dns
	process table	Any TCP			Xterm	Any user session	sendmail	smtp
	smurf	icmp	-		-		xlock	X
	syslogd	syslog					xsnoop	X
	teardrop	N/A					-	-
	udpstorm	echo/chargen						

The NSL-KDD dataset consists of two parts: (i) KDDTrain$^+$ and (ii) KDDTest$^+$. The KDDTrain$^+$ part of the NSL-KDD dataset is used to train a detection method or system to detect network intrusions. It contains four classes of attacks and a normal class dataset. The KDDTest$^+$ part of NSL-KDD dataset is used for testing a detection method or a system when it is evaluated for performance. It also contains the same classes of traffic present in the training set. The distribution of attack and normal instances in the NSL-KDD dataset is shown in Table 3.

Table 3: Distribution of normal and attack traffic instances in NSL-KDD dataset

Dataset	*DoS*	*u2r*	*r2l*	*Probe*	*Normal*	*Total*
KDDTrain$^+$	45927	52	995	11656	67343	125973
KDDTest$^+$	7458	67	2887	2422	9710	22544

2.2.3 DARPA 2000 Dataset

A DARPA[2] evaluation project [18] targeted the detection of complex attacks that contain multiple steps. Two attack scenarios were simulated in the DARPA 2000 evaluation contest, namely Lincoln Laboratory scenario DDoS (LLDOS) 1.0 and LLDOS 2.0. To achieve variations, these two attack scenarios were carried out over several network and audit scenarios. These sessions were grouped into four attack phases: (a) probing, (b) breaking into the system by exploiting vulnerability, (c) installing DDoS software for the compromised system, and (d) launching DDoS attack against another target. LLDOS 2.0 is different from LLDOS 1.0 in that attacks are more stealthy and thus harder to detect. Since this dataset contains multi-stage attack scenarios, it is also commonly used for evaluation of alert correlation techniques.

2.2.4 DEFCON Dataset

The DEFCON[3] dataset is another commonly used dataset for evaluation of IDSs [11]. It contains network traffic captured during a hacker competition called Capture The Flag (CTF), in which competing teams are divided into two groups: *attackers* and *defenders*. The traffic produced during CTF is very different from real world network traffic since it contains only intrusive traffic without any normal background traffic. Due to this limitation, DEFCON dataset has been found useful only in evaluating alert correlation techniques.

2.2.5 CAIDA Dataset

CAIDA[4] collects many different types of data and makes them available to the research community. CAIDA datasets [8] are very specific to particular events or attacks. Most of its longer

[2] http://www.ll.mit.edu/mission/communications/ist/corpora/ideval/data/index.html

[3] http://cctf.shmoo.com/data/

[4] http://www.caida.org/home/

traces are anonymized backbone traces without their payload. The CAIDA DDoS 2007 attack dataset contains one hour of anonymized traffic traces from DDoS attacks on August 4, 2007, which attempted to consume a large amount of network resources when connecting to Internet servers. The traffic traces contain only attack traffic to the victim and responses from the victim with 5 minutes split form. All traffic traces are in pcap (tcpdump) format. The creators removed non-attack traffic as much as possible when creating the CAIDA DDoS 2007 dataset.

2.2.6 LBNL Dataset

LBNL's internal enterprise traffic traces are full header network traces without payload [23]. This dataset suffers from heavy anonymization to the extent that scanning traffic was extracted and separately anonymized to remove any information which could identify individual IPs. The background and attack traffic in the LBNL dataset are described below.

- *LBNL background traffic:* This dataset can be obtained from the Lawrence Berkeley National Laboratory (LBNL) in the US. Traffic in this dataset is comprised of packet level incoming, outgoing and internally routed traffic streams at the LBNL edge routers. Traffic was anonymized using the $tcpmkpub$ tool [35]. The main applications observed in the internal and external traffic are Web, email and name services. Other applications like Windows services, network file services and backup were used by internal hosts. The details of each service and information on each packet and other relevant description are given in [34]. The background network traffic statistics of the LBNL dataset are given in Table 4.

- *LBNL attack traffic:* This dataset identifies attack traffic by isolating scans in aggregate traffic traces. Scans are identified by flagging those hosts which unsuccessfully probe more than 20 hosts, out of which 16 hosts are probed in ascending or descending IP order [35]. Malicious traffic mostly consists of failed incoming TCP SYN requests, i.e., TCP port scans targeted towards LBNL hosts. However, there are also some outgoing TCP scans in the dataset. Most UDP traffic observed in the data (incoming and outgoing) is comprised of successful connections, i.e., host replies for the received UDP flows. Clearly, the attack rate is significantly lower than the background traffic rate. Details of the attack traffic in this dataset are shown in Table 4. Complexity and privacy were two main reservations of the participants of the endpoint data collection study. To address these reservations, the dataset creators developed a custom multi-threaded MS Windows tool using the $Winpcap$ API [7] for data collection. To reduce packet logging complexity at the endpoints, they only logged very elementary session-level information (bidirectional communication between two IP addresses on different ports) for the TCP and UDP packets. To ensure user privacy, an anonymization policy was used to anonymize all traffic instances.

2.2.7 Endpoint Dataset

The background and attack traffic for the endpoint datasets are described below.

- *Endpoint background traffic:* In the endpoint context, we see in Table 5 that home computers generate significantly higher traffic volumes than office and university computers because: (*i*) they are generally shared between multiple users, and (*ii*) they run peer-to-peer and multimedia applications. The large traffic volumes of home computers are also evident from their high mean number of sessions per second. To generate attack traffic, developers on infect Virtual Machines (VMs) at the endpoints with different malware, viz., Zotob.G, Forbot-FU, Sdbot-AFR, Dloader-NY, So-Big.E@mm, MyDoom.A@mm, Blaster, Rbot-AQJ, and RBOT.CCC. Details of the malware can be found in [42]. Characteristics of the attack traffic in this dataset are given in Table 6. These malwares have diverse scanning rates and attack ports or applications.

- *Endpoint attack traffic:* The attack traffic logged at the endpoints is mostly comprised of outgoing port scans. Note that this is the opposite of the LBNL dataset, in which most attack traffic is inbound. Moreover, the attack traffic rates at the endpoints are generally much higher than the background traffic rates of the LBNL datasets. This diversity in attack direction and rates provides a sound basis for performance comparison among scan detectors. For each malware, attack traffic of 15 minute duration was inserted in the background traffic for each endpoint at a random time instance. This operation was repeated to insert 100 non-overlapping attacks of each worm inside each endpoint's background traffic.

2.3 Real-life Datasets

We discuss three real-life datasets created by collecting network traffic on several consecutive days. The details include both normal as well as attack traffic in appropriate proportions in the authors' respective campus networks (i.e., testbeds).

2.3.1 UNIBS Dataset

The UNIBS packet traces [45] were collected on the edge router of the campus network of the University of Brescia in Italy, on three consecutive working days. The dataset includes traffic captured or collected and stored using 20 workstations, each running the GT (Ground Truth) client daemon. The dataset creators collected the traffic by running tcpdump on the faculty router, which was a dual Xeon Linux box that connected the local network to the Internet through a dedicated 100Mb/s uplink. They captured and stored the traces on a dedicated disk of a workstation connected to the router through a dedicated ATA controller.

2.3.2 ISCX-UNB Dataset

The ISCX-UNB dataset [37] is built on the concept of profiles that include the details of intrusions. The datasets were col-

Table 4: Background and attack traffic information for the LBNL datasets

Date	*Duration (mins)*	*LBNL hosts*	*Remote hosts*	*Background traffic rate (packet/sec)*	*Attack traffic rate (packet/sec)*
10/04/2004	10 min	4,767	4,342	8.47	0.41
12/15/2004	60 min	5,761	10,478	3.5	0.061
12/16/2004	60 min	5,210	7,138	243.83	72

Table 5: Background traffic information for four endpoints with high and low rates

Endpoint ID	*Endpoint type*	*Duration (months)*	*Total sessions*	*Mean session rate (/sec)*
3	Home	3	3,73,009	1.92
4	home	2	4,44,345	5.28
6	University	9	60,979	0.19
10	University	13	1,52,048	0.21

Table 6: Endpoint attack traffic for two high and two low-rate worms

Malware	*Release Date*	*Avg. Scan rate (/sec)*	*Port (s) Used*
Dloader-NY	Jul 2005	46.84 sps	TCP 1,35,139
Forbot-FU	Sept 2005	32.53 sps	TCP 445
Rbot-AQJ	Oct 2005	0.68 sps	TCP 1,39,769
MyDoom-A	Jan 2006	0.14 sps	TCP 3127-3198

lected using a real-time testbed by incorporating multi-stage attacks. It uses two profiles - α and β - during the generation of the datasets. α profiles are constructed using the knowledge of specific attacks and β profiles are built using the filtered traffic traces. Real packet traces were analyzed to create α and β profiles for agents that generate real-time traffic for HTTP, SMTP, SSH, IMAP, POP3 and FTP protocols. Various multistage attack scenarios were explored to generate malicious traffic.

2.3.3 KU Dataset

The Kyoto University dataset[5] is a collection of network traffic data obtained from honeypots. The raw dataset obtained from the honeypot system consisted of 24 statistical features, out of which 14 significant features were extracted [38]. The dataset developers extracted 10 additional features that could be used to investigate network events inside the university more effectively. The initial 14 features extracted are similar to those in the KDDcup99 datasets. Only 14 conventional features were used during training and testing.

2.4 Discussion

The datasets described above are valuable assets for the intrusion detection community. However, the benchmark datasets suffer from the fact that they are not good representatives of real world traffic. For example, the DARPA dataset has been questioned about the realism of the background traffic [27, 29] because it is synthetically generated. In addition to the difficulty of simulating real time network traffic, there are additional challenges in IDS evaluation [30]. These include difficulties in collecting attack scripts and victim software, differing requirements for testing signature based vs. anomaly based IDSs, and host-based vs. network based IDSs. In addition to these, we make the following observations based on our analysis.

- Most datasets are not labelled properly due to non-availability of actual attack information. These include KDDcup99, UNIBS, Endpoint and LBNL datasets.
- The proportion of normal and attack ratios are different in different datasets [21, 38, 45].
- Several existing datasets [21, 23, 38] have not been maintained or updated to reflect recent trends in network traffic by incorporating evolved network attacks.
- Most existing datasets are annonymized [8, 18] due to potential security risks to an organization. They do not share their raw data with researchers.
- Several datasets [8, 23, 18, 45] lack in traffic features. They have only raw traffic traces but it is important to extract relevant traffic features for individual attack identification.

3 Real-life Datasets Generation

As noted above, the generation of an unbiased real-life intrusion dataset incorporating a large number of real world attacks is important to evaluate network anomaly detection methods and systems. In this paper, we describe the generation of three real-life network intrusion datasets[6] including (a) a TUIDS (Tezpur University Intrusion Detection System) intrusion dataset, (b) a TUIDS coordinated scan dataset, and (c) a TUIDS DDoS dataset at both packet and flow levels [16]. The resulting details and supporting infrastructure is discussed in the following subsections.

3.1 Testbed Network Architecture

The TUIDS testbed network consists of 250 hosts, 15 L2 switches, 8 L3 switches, 3 wireless controllers, and 4 routers that compose 5 different networks inside the Tezpur University campus. The architectures of the TUIDS testbed and TUIDS testbed for DDoS dataset generation are given in Figures 2 and

[5] http://www.takakura.com/kyoto_data

[6] http://agnigarh.tezu.ernet.in/∼dkb/resources.html

3, respectively. The hosts are divided into several VLANs, each VLAN belonging to an L3 switch or an L2 switch inside the network. All servers are installed inside a DMZ[7] to provide an additional layer of protection in the security system of an organization.

3.2 Network Traffic Generation

To generate real time normal and attack traffic, we configured several hosts, workstations, and servers in the TUIDS testbed network. The network consists of 6 interconnected Ubuntu 10.10 workstations. On each workstation, we have installed several severs including a network file server (Samba), a mail sever (Dovecot), a telnet server, an FTP server, a Web server, and an SQL sever with PHP compatibility. We also installed and configured 4 Windows Servers 2003 to exploit a diverse set of known vulnerabilities against the testbed environment. Servers and their services running on our testbed are summarized in Table 7.

Table 7: Servers and their services running on the testbed network

Server	*Operating system*	*Services*	*Provider*
Main Server	Ubuntu 10.10	Web, eMail	Apache 2.4.3, Dovecot 2.1.14
Network File Server	Ubuntu 10.10	Samba	Samba 4.0.2
Telnet Server	Ubuntu 10.10	Telnet	telnet-0.17-36bulid1
FTP Server	Ubuntu 10.10	ftp	vsFTPd 2.3.0
Windows Server	Windows Server 2003	Web	IIS v7.5
MySQL Server	Ubuntu 10.10	database	MySQL 5.5.30

The normal network traffic is generated based on the day-to-day activities of users and especially generated traffic from configured servers. It is important to generate different types of normal traffic. So, we capture traffic from students, faculty members, system administrators, and office staff on different days within the University. The attack traffic is generated by launching attacks within the testbed network in three different subsets, viz., a TUIDS intrusion dataset, a coordinated scan dataset and a DDoS dataset. The attacks launched in the generation of these real-life datasets are summarized in Table 8.

As seen in the table above, 22 distinct attack types (1-22 in Table 8) were used to generate the attack traffic for the TUIDS intrusion dataset; six attacks (17-22 in Table 8) were used to generate the attack traffic for the coordinated scan dataset and finally six attacks (23-28 in Table 8) were used to generate the attack traffic for a DDoS dataset with combination of TCP, UDP and ICMP protocols.

[7]Demilitarized zone is a network segment located between a secured local network and unsecured external networks (Internet). A DMZ usually contains servers that provide services to users on the external network, such as Web, mail and DNS servers that are hardened systems. Typically, two firewalls are installed to form the DMZ.

Table 8: List of real time attacks and their generation tools

Attack name	*Generation tool*	*Attack name*	*Generation tool*
1.bonk	targa2.c	15.linux-icmp	linux-icmp.c
2.jolt	targa2.c	16.syn-flood	synflood.c
3.land	targa2.c	17.window-scan	nmap/rnmap
4.saihyousen	targa2.c	18.syn-scan	nmap/rnmap
5.teardrop	targa2.c	19.xmasstree-scan	nmap/rnmap
6.newtear	targa2.c	20.fin-scan	nmap/rnmap
7.1234	targa2.c	21.null-scan	nmap/rnmap
8.winnuke	targa2.c	22.udp-scan	nmap/rnmap
9.oshare	targa2.c	23.syn-flood(DDoS)	LOIC
10.nestea	targa2.c	24.rst-flood(DDoS)	Trinity v3
11.syndrop	targa2.c	25.udp-flood(DDoS)	LOIC
12.smurf	smurf4.c	26.ping-flood(DDoS)	DDoS ping v2.0
13.opentear	opentear.c	27.fraggle udp-flood(DDoS)	Trinoo
14.fraggle	fraggle.c	28.smurf icmp-flood(DDoS)	TFN2K

3.3 Attack Scenarios

The attack scenarios start with information gathering techniques collecting target network IP ranges, identities of name servers, mail servers and user e-mail accounts, etc. This is achieved by querying the DNS for resource records using network administrative tools like nslookup and dig. We consider six attack scenarios when collecting real time network traffic for dataset generation.

3.3.1 *Scenario 1*: Denial of Service Using Targa

This attack scenario is designed to perform attacks on a target using the targa[8] tool until it is successful. Targa is a very powerful tool to quickly damage a particular network belonging to an organization. We ran targa by specifying different parameter values such as IP ranges, attacks to run and number of times to repeat the attack.

3.3.2 *Scenario 2*: Probing Using nmap

In this scenario, we attempt to acquire information about the target host and then launch the attack by exploiting the vulnerabilities found using the nmap[9] tool. Examples of attacks that can be launched by this method are syn-scan and ping-sweep.

3.3.3 *Scenario 3*: Coordinated Scan Using rnmap

This scenario starts with a goal to perform coordinated port scans to single and multiple targets. Tasks are distributed

[8]http://packetstormsecurity.com/
[9]http://nmap.org/

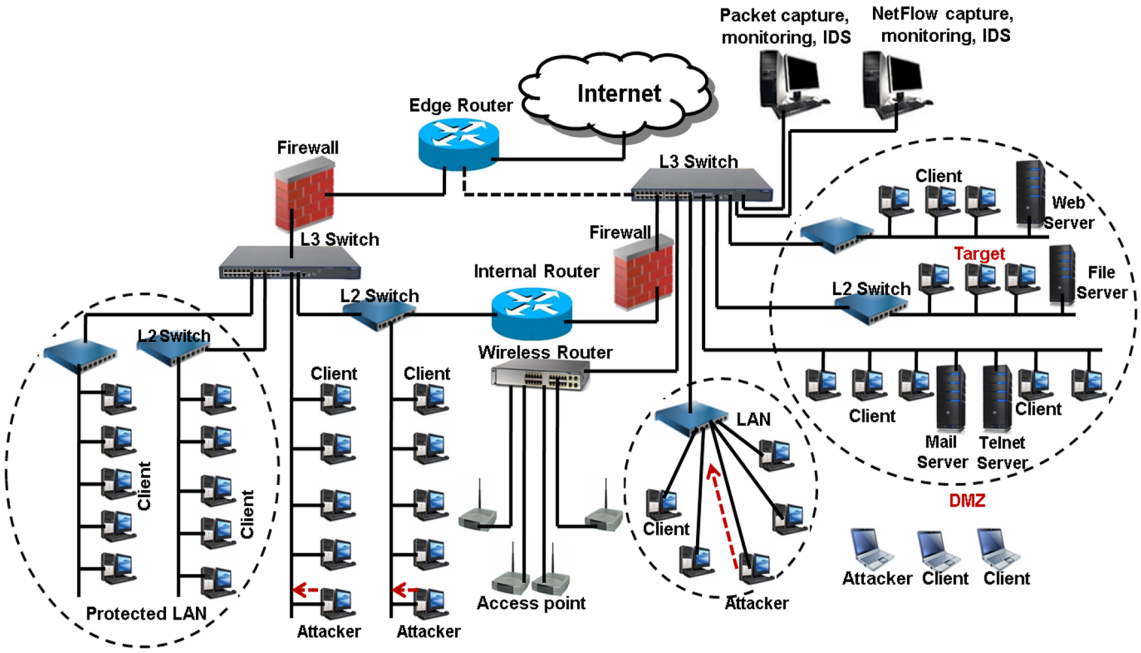

Figure 2: Testbed network architecture used during TUIDS dataset generation

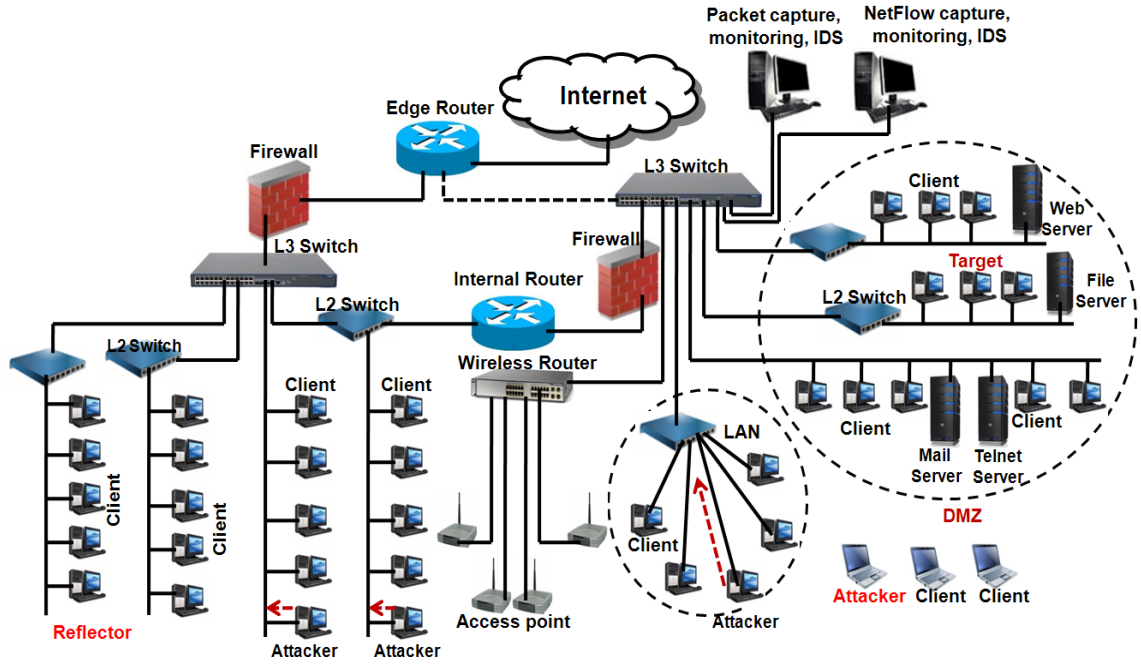

Figure 3: Testbed network architecture used during DDoS dataset generation

among multiple hosts for individual actions which may be synchronized. We use the rnmap[10] tool to launch coordinated scans in our testbed network during the collection of traffic.

3.3.4 *Scenario 4*: User to Root Using Brute Force ssh

These attacks are very common against networks as they tend to break into accounts with weak username and password combinations. This attack has been designed with the goal of acquiring an SSH account by running a brute force dictionary attack against our central server. We use the brutessh[11] tool and a customized dictionary list. The dictionary consists of over 6100 alphanumeric entries of varying length. We executed the attack for 60 minutes, during which superuser credentials were returned from the server. This ID and password combination was used to download other users' credentials immediately.

[10]http://rnmap.sourceforge.net/

[11]http://www.securitytube-tools.net/

3.3.5 *Scenario 5*: Distributed Denial of Service Using Agent-handler Network

This scenario mainly attempts to exploit an agent handler network to launch the DDoS attack in the TUIDS testbed network. The agent-handler network consists of clients, handlers and agents. The handlers are software packages that are used by the attacker to communicate indirectly with the agents. The agent software exists in compromised systems that will eventually carry out the attack on the victim system. The attacker may communicate with any number of handlers, thus making sure that the agents are up and running. We use Trinity v3, TFN2K, Trinoo, and DDoS ping 2.0 to launch the attacks in our testbed.

3.3.6 *Scenario 6*: Distributed Denial of Service Using IRC Botnet

Botnets are an emerging threat to all organizations because they can compromise a network and steal important information and distribute malware. Botnets combine individual malicious behaviors into a single platform by simplifying the actions needed to be performed by users to initiate sophisticated attacks against computers or networks around the world. These behaviors include coordinated scanning, DDoS activities, direct attacks, indirect attacks and other deceitful activities taking place across the Internet.

The main goal of this scenario is to perform distributed attacks using infected hosts on the testbed. An Internet Relay Chat (IRC) bot network allow users to create public, private and secret channels. For this, we use a LOIC[12], an IRC-based DDoS attack generation tool. The IRC systems have several other significant advantages for launching DDoS attacks. Among the three important benefits are (i) they afford a high degree of anonymity, (ii) they are difficult to detect, and (iii) they provide a strong, guaranteed delivery system. Furthermore, the attacker no longer needs to maintain a list of agents, since he can simply log on to the IRC server and see a list of all available agents. The IRC channels receive communications from the agent software regarding the status of the agents (i.e., up or down) and participate in notifying the attackers regarding the status of the agents.

3.4 Capturing Traffic

The key tasks in network traffic monitoring are lossless packet capturing and precise timestamping. Therefore, software or hardware is required with a guarantee that all traffic is captured and stored. The real network traffic is captured using the *Libpcap* [19, 20] library, an open source C library offering an interface for capturing link-layer frames over a wide range of system architectures. It provides a high-level common Application Programming Interface (API) to the different packet capture frameworks of various operating systems. The offered abstraction layer allows programmers to rapidly develop highly portable applications. A hierarchy of network traffic capturing components is given in Figure 4 [10].

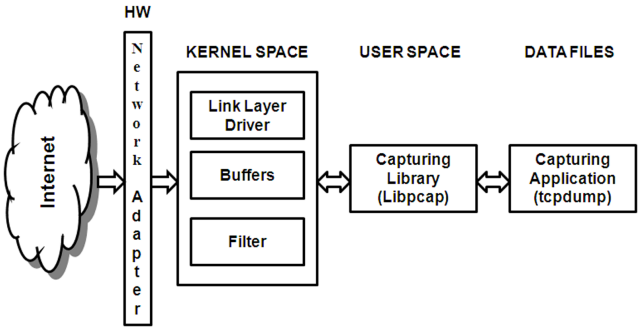

Figure 4: Hierarchy of Network Traffic Capturing Components

Libpcap defines a common standard format for files in which captured frames are stored, also known as the *tcpdump* format, currently a de facto standard used widely in public network traffic archives. Modern kernel-level capture frameworks on UNIX operating systems are mostly based on the BSD (or Berkeley) Packet Filter (BPF) [28]. The BPF is a software device that taps network interfaces, copying packets into kernel buffers and filtering out unwanted packets directly in interrupt context. Definitions of packets to be filtered can be written in a simple human readable format using Boolean operators and can be compiled into a pseudo-code to be passed to the BPF device driver by a system call. The pseudo-code is interpreted by the BPF Pseudo-Machine, a lightweight, high-performance, state machine specifically designed for packet filtering. *Libpcap* also allows programmers to write applications that transparently support a rich set of constructs to build detailed filtering expressions for most network protocols. A few *Libpcap* system calls can be read directly from user's command line, compile into pseudo-code and passed it to the Berkeley Packet Filter. *Libpcap* and the BPF interact to allow network packet data to traverse several layers to finally be processed and transformed into capture files (i.e., *tcpdump* format) or samples for statistical analysis.

With the goal of preparing both packet and flow level datasets, we capture both packet and NetFlow traffic from different locations in the TUIDS testbed. The capturing period started at 08:00:05 am on Monday February 21, 2011 and continuously ran for an exact duration of seven days, ending at 08:00:05 am on Sunday February 27th. Attacks were executed during this period for the TUIDS intrusion and the coordinated scan datasets. DDoS traffic was also collected for the same amount of time but during October, 2012 with several variations of real time DDoS attacks. Figure 5 illustrates the protocol composition and the average throughput during the last hour of data capture for the TUIDS intrusion dataset.

We use a tool known as lossless gigabit remote packet capture with Linux (Gulp[13]) for capturing packet level traffic in a mirror port as shown in the TUIDS testbed architecture. Gulp reads packets directly from the network card and writes to the disk at a high rate of packet capture without dropping packets. For low-rate packets, Gulp flushes the ring buffer if it has not written anything in the last second. Gulp writes into

[12]http://sourceforge.net/projects/loic/

[13]http://staff.washington.edu/corey/gulp/

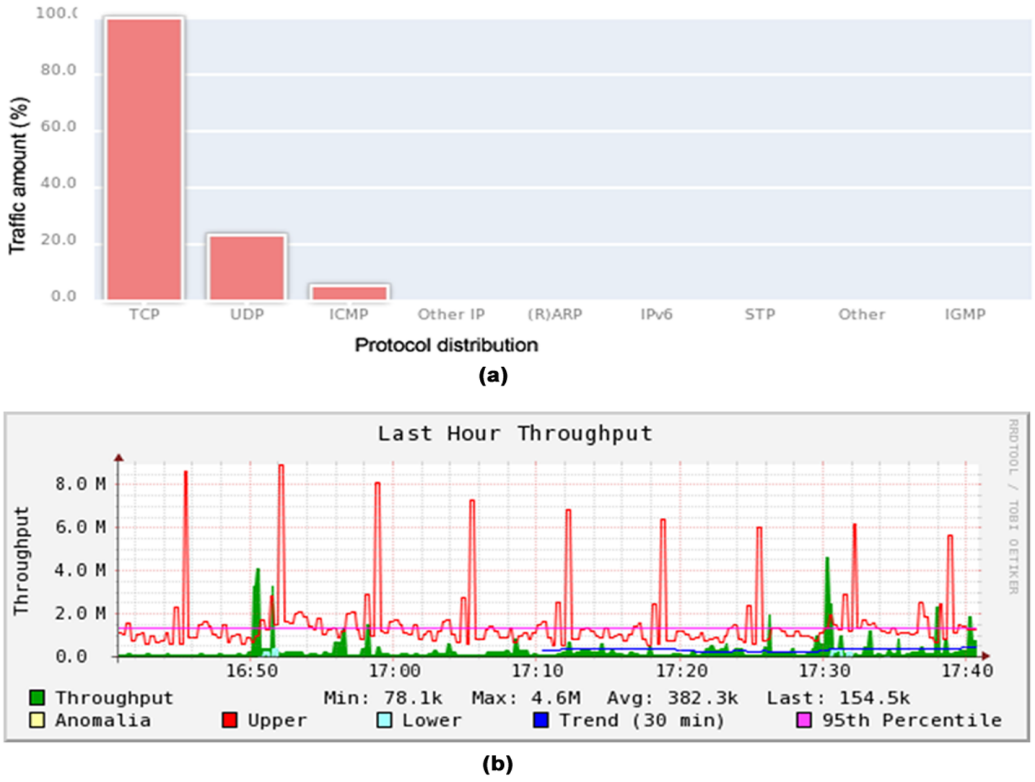

Figure 5: (a) composition of protocols and (b) average throughput during last hour of data capture for the TUIDS intrusion dataset seen in our lab's traffic

even block boundaries for excellent writing performance when the data rate increases. It stops filling the ring buffer after receiving an interrupt but it would write into the disk whatever remains in the ring buffer.

In the last few years, NetFlow has become the most popular approach for IP network monitoring, since it helps cope with scalability issues introduced by increasing network speeds. Now major vendors offer flow-enabled devices, such as Cisco routers with NetFlow. A NetFlow is a stream of packets that arrives on a source interface with the key values shown in Figure 6. A key is an identified value for a field within the packet. Cisco routers have NetFlow features that can be enabled to generate NetFlow records. The principle of NetFlow is as follows: When the router receives a packet, its NetFlow module scans the source IP address, the destination IP address, the source port number, the destination port number, the protocol type, the type of service (ToS) bit in the IP header, and the input or output interface number on the router of the IP packet to judge whether it belongs to a NetFlow record that already exists in the cache. If so, it updates the NetFlow record; otherwise, a new NetFlow record is generated in the cache. The expired NetFlow records in the cache are exported periodically to a destination IP address using a UDP port.

For capturing the NetFlow traffic, we need a NetFlow collector that can listen to a specific UDP port for getting traffic. The NetFlow collector captures exported traffic from multiple routers and periodically stores it in summarized or aggregated format into a round robin database (RRD). The following tools are used to capture and visualize the NetFlow traffic.

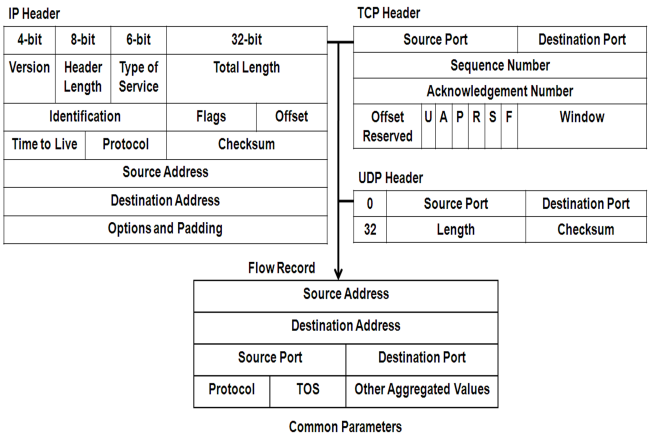

Figure 6: Common NetFlow parameters

(a) *NFDUMP*: This tool captures and displays NetFlow traffic. All versions of nfdump support NetFlow v5, v7, and v9. nfcapd is a NetFlow capture daemon that reads the NetFlow data from the routers and stores the data into files periodically. It automatically rotates files every n minutes (by default it is 5 minutes). We need one nfcapd process for each NetFlow stream. Nfdump reads the NetFlow data from the files stored by nfcapd. The syntax is similar to that of tcpdump. Nfdump displays NetFlow data and can create top N statistics for flows based on the parameters selected. The main goal is to analyze NetFlow data from the past as well as to track interesting traffic

Table 9: Parameters identified for packet level data

Sl. No.	*Parameter name*	*Description*
1	Time	Time since occurrence of first frame
2	Frame No	Frame number
3	Frame Len	Length of a frame
4	Capture Len	Capture length
5	TTL	Time to live
6	Protocol	Protocols (such as, TCP, UDP, ICMP etc.)
7	Src IP	Source IP address
8	Dst IP	Destination IP address
9	Src port	Source port
10	Dst port	Destination port
11	Len	Data length
12	Seq No	Sequence number
13	Header Len	Header length
14	CWR	Congestion window record
15	ECN	Explicit congestion notification
16	URG	Urgent TCP flag
17	ACK	Acknowledgement flag
18	PSH	Push flag
19	RST	Reset flag
20	SYN	TCP syn flag
21	FIN	TCP fin flag
22	Win Size	Window Size
23	MSS	Maximum segment size

patterns continuously from high speed networks. The amount of time from the past is limited only by the disk space available for all NetFlow data.

Nfdump has four fixed output formats: *raw*, *line*, *long* and *extended*. In addition, the user may specify any desired output format by customizing it. The default format is line, unless specified. The raw format displays each record in multiple lines and prints any available information in the traffic record. (b) *NFSEN*: NfSen is a graphical Web based front end tool for visualization of NetFlow traffic. NfSen facilitates the visualization of several traffic statistics, e.g., flow-wise statistics for various features, navigation through the NetFlow traffic, processes within a time span and continuous profiles. It can also add own plugins to process NetFlow traffic in a customized manner at a regular time interval.

Normal traffic is captured by restricting it to the internal networks, where 80% of the hosts are connected to the router, including wireless networks. We assume that normal traffic follows the normal probability distribution. Attack traffic is captured as we launch various attacks in the testbed for a week. For DDoS attacks, we used packet-craft[14] to generate customized packets. Figures 7 and 8 show the number of flows per second and also the protocol-wise distribution of flows during the capturing period, respectively.

3.5 Feature Extraction

We use wireshark and Java routines for filtering unwanted packets (such as packets with routing protocols, and packets with application layer protocols) as well as irrelevant information from captured packets. Finally, we retrieve all relevant information from each packet using Java routines and store it in comma separated form in a text file. The details of parameters identified for packet level data are shown in Table 9.

We developed several C routines and used them for filtering NetFlow data and for extracting features from the captured data. A detailed list of parameters identified for flow level data

[14]http://www.packet-craft.net/

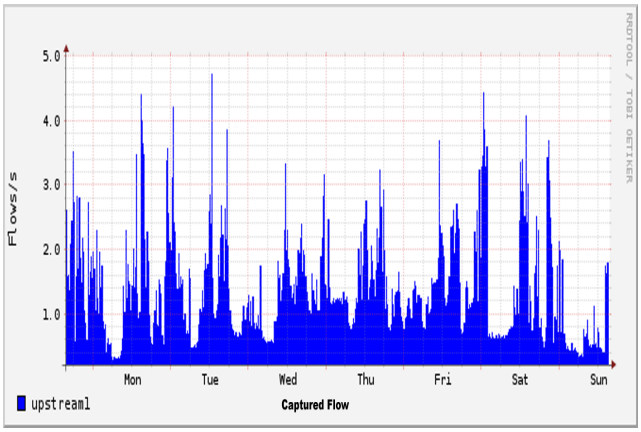

Figure 7: Number of flows per second in TUIDS intrusion datasets during the capture period

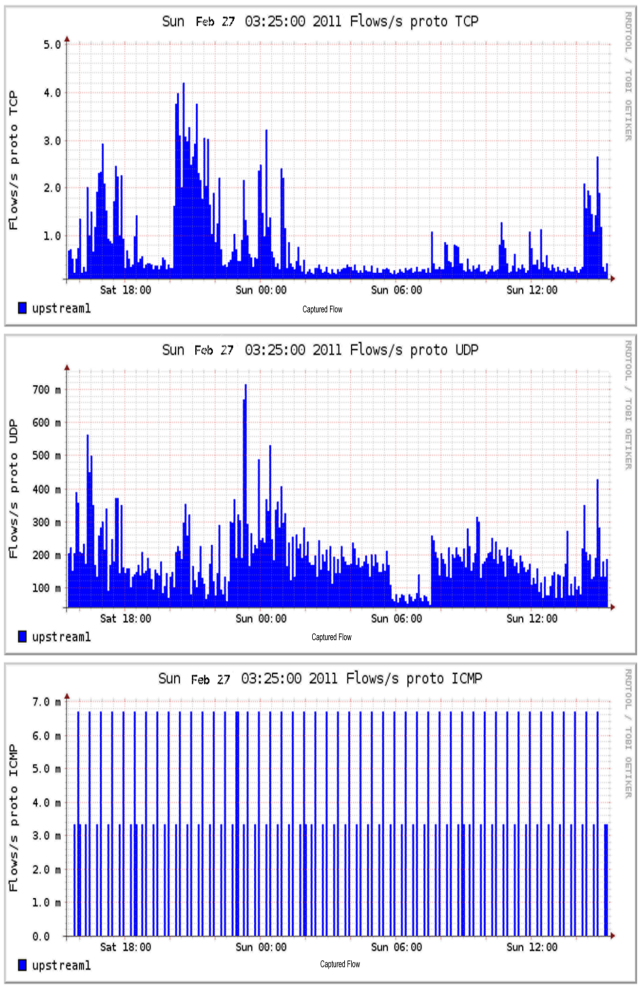

Figure 8: Protocol-wise distribution of flow per second in TUIDS intrusion dataset during the capture period

is given in Table 10.

We capture, preprocess and extract various features in both packet and flow level network traffic. We introduce a framework for fast distributed feature extraction from raw network traffic, correlation computation and data labelling, as shown in Figure 9. We extract four types of features: *basic*, *content* based, *time* based and *connection* based, from the raw network traffic. We use $T = 5\ seconds$ as the time window for extraction of both time based and connection based traffic features. S_1 and S_2 are servers used for preprocessing, attack labelling and profile generation. WS_1 and WS_2 are high-end workstations used for basic feature extraction and merging packet and NetFlow traffic. $N_1, N_2, \cdots N_6$ are independent nodes used for protocol specific feature extraction. The lists of extracted features at both packet and flow levels for the intrusion datasets are presented in Table 11 and Table 12, respectively. The list of features available in the KDDcup99 intrusion dataset is also shown in Table 13.

Table 10: Parameters identified for flow level data

Sl. No.	*Parameter name*	*Description*
1	flow-start	Starting of flow
2	Duration	Total life time of a flow
3	Proto	Protocol, i.e., TCP, UDP, ICMP etc.
3	Src-IP	Source IP address
4	Src-port	Source port
5	Dest-IP	Destination IP address
6	Dest-port	Destination port
7	Flags	TCP flags
8	ToS	Type of Service
9	Packets	Packets per flow
10	Bytes	Bytes per flow
11	Pps	Packet per second
12	Bps	Bit per second
13	Bpp	Byte per packet

3.6 Data Processing and Labelling

As reported in the previous section, traffic features are extracted separately (within a time interval). So, it is important to correlate each feature (i.e., basic, content based, time based, and connection based) to a time interval. Once correlation is performed for both packet and flow level traffic, labelling of each feature data as normal or anomalous is important. The labelling process enriches the feature data with information such as (i) the type and structure of malicious or anomalous data, and (ii) dependencies among different isolated malicious activities. The correlation and labelling of each feature traffic as normal or anomalous is made using Algorithm 1. $F = \{\alpha, \beta, \gamma, \delta\}$ is the set of extracted features, where α is the set of basic features, β is the set of content-based features, γ is the set of time-based features and δ is the set of connection-based features. Both normal and anomalous traffic are collected separately in several sessions within a week. We remove normal traffic from anomalous traces as much as possible.

The overall traffic composition with protocol distribution in the generated datasets is summarized in Table 14. The traffic includes the TUIDS intrusion dataset, the TUIDS coordinated scan dataset and the TUIDS DDoS dataset. The final labelled feature datasets for each category with the distribution of normal and attack information are summarized in Table 15. All datasets are prepared at both packet and flow levels and presented in terms of training and testing in Table 15.

Algorithm 1 : FC and labelling (F)

Input: extracted feature set, $F = \{\alpha, \beta, \gamma, \delta\}$
Output: correlated and labelled feature data, D

initialize D
call $FeatureExtraction()$, $F \leftarrow \{\alpha, \beta, \gamma, \delta\}$, ▷ the procedure FeatureExtraction() extracts the features separately for all cases
for $i \leftarrow 1$ to $|N|$ **do** ▷ N is the total traffic instances
  for $i \leftarrow 1$ to $|F|$ **do** ▷ F is the total traffic features
    if $(unique(src.ip \wedge dst.ip))$ **then**
      store $D[ij] \leftarrow \alpha_{ij}, \beta_{ij}$
    end if
    if $((T == 5s) \wedge (LnP == 100))$ **then** ▷ T is the time window, LnP is the last n packets
      Store $D[ij] \leftarrow \gamma_{ij}, \delta ij$
    end if
  end for
  $D[ij] \leftarrow \{normal, attack\}$ ▷ label each traffic feature instance based on the duration of the collected traffic
end for

3.7 Comparison with Other Public Datasets

Several real network traffic traces are readily available to the research community as reported in Section 2. Although these traffic traces are invaluable to the research community most if not all, fail to satisfy one or more requirements described in Section 1. This paper is mostly distinguished by the fact that the issue of data generation is approached from what other datasets have been unable to provide, for the network security community. It attempts to resolve the issues seen in other datasets by presenting a systematic approach to generate real-life network intrusion datasets. Table 16 summarizes a comparison between the prior datasets and the dataset generated through the application of our systematic approach to fulfill the principal objectives outlined for qualifying dataset.

Most datasets are unlabelled as labelling is labor-intensive and requires a comprehensive search to tag anomalous traffic. Although an IDS helps by reducing the work, there is no guarantee that all anomalous activity is labelled. This has been a major issue with all datasets and one of the reasons behind the post insertion of attack traffic in the DARPA 1999 dataset, so that anomalous traffic can be labelled in a deterministic manner. Having seen the inconsistencies produced by traffic merging, this paper has adopted a different approach to provide the same level of deterministic behavior with respect to anomalous traffic by conducting anomalous activity within the capturing period using available network resources. Through the use of logging, all ill-intended activity can be effectively labelled.

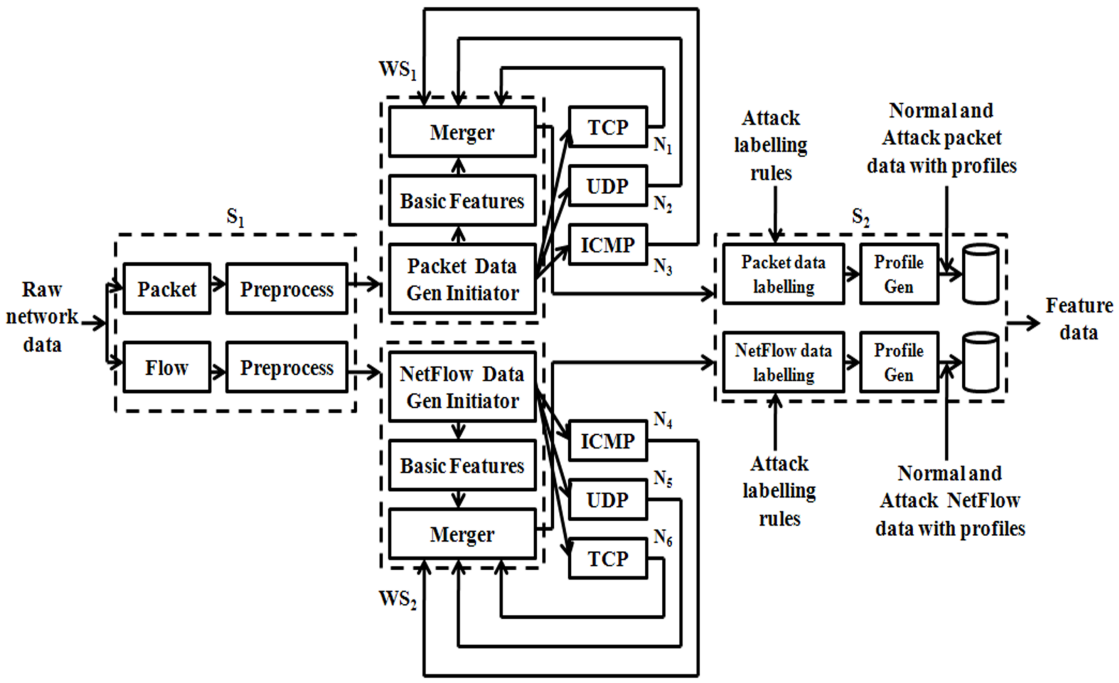

Figure 9: Fast distributed feature extraction, correlation and labelling framework

The extent and scope of network traffic capture become relevant in situations where the information contained in the traces may breach the privacy of individuals or organizations. In order to prevent privacy issues, almost all publicly available datasets remove any identifying information such as payload, protocol, destination and flags. In addition, the data is anonymized where necessary header information is cropped or flows are just summarized.

In addition to anomalous traffic, traces must contain background traffic. Most captured datasets have little control over the anomalous activities included in the traces. However, a major concern with evaluating anomaly based detection approaches is the requirement that anomalous traffic must be present at a certain scale. Anomalous traffic also tends to become outdated with the introduction of more sophisticated attacks. So, we have generated more up-to-date datasets that reflect the current trends and are tailored to evaluate certain characteristics of detection mechanisms which are unique to themselves.

As discussed earlier, several datasets are available for evaluating an IDS. Network intrusion detection researchers evaluate detection methods using intrusion datasets to demonstrate how their methods can handle recent attacks and network environments. We have used our datasets to evaluate several network intrusion detection methods. Some of them are outlier-based network anomaly detection approach (NADO) [4], an unsupervised method [3, 6], an adaptive outlier-based coordinated scan detection approach (AOCD) [5], and a multi-level hybrid IDS (MLH-IDS) [15]. We found better results in almost all the experiments when we used TUIDS dataset in terms of false positive rate, true positive rate and F-measure.

3.8 Comparison with Other Relevant Work

Our approach differs from other works as follows.

- The NSL-KDD [32] dataset is an enhanced version of the KDDcup99 intrusion dataset prepared by Tavallaee et al. [43]. This dataset is too old to evaluate a modern detection method or a system that has been developed recently. It removes repeated traffic records from the old KDDcup99 dataset. In contrast, our datasets are prepared using diverse attack scenarios incorporating recent attacks. Our datasets contain both packet and flow level information that help detect attacks more effectively in high speed networks.

- Song et al. [39] prepared the KU dataset and used the dataset to evaluate an unsupervised network anomaly detection method. This dataset contains 17 different features at packet level only. In contrast, we present a systematic approach to generate real-life network intrusion datasets and prepared three different categories of datasets at both packet and flow levels.

- Like Shiravi et al. [37], our approach considers recently developed attacks and attacks on network layers when generating the datasets. Shiravi et al. concentrate mostly on application-layer attacks. They build profiles for different real-world attack scenarios and use them to generate traffic that follows the same behavior while generating the dataset at packet level. In comparison, we generate three different categories of datasets at both packet and flow levels for the research community to evaluate detection methods or systems. Since we have extracted more number of features at both packet and flow levels. Our

Table 11: List of packet level features in TUIDS intrusion dataset

Label/feature name	*Type*	*Description*
Basic features		
1. Duration	C	Length (number of seconds) of the connection
2. Protocol-type	D	Type of protocol, e.g., tcp, udp, etc.
3. Src-ip	C	Source host IP address
4. Dest-ip	C	Destination IP address
5. Src-port	C	Source host port number
6. Dest-port	C	Destination host port number
7. Service	D	Network service at the destination, e.g., http, telnet, etc.
8. num-bytes-src-dst	C	The number of data bytes flowing from source to destination
9. num-bytes-dst-src	C	The number of data bytes flowing from destination to source
10. Fr-no	C	Frame number
11. Fr-len	C	Frame length
12. Cap-len	C	Captured frame length
13. Head-len	C	Header length of the packet
14. Frag-off	D	Fragment offset: '1' for the second packet overwrite everything, '0' otherwise
15. TTL	C	Time to live: '0' discards the packet
16. Seq-no	C	Sequence number of the packet
17. CWR	D	Congestion window record
18. ECN	D	Explicit congestion notification
19. URG	D	Urgent TCP flag
20. ACK	D	Acknowledgement flag value
21. PSH	D	Push TCP flag
22. RST	D	Reset TCP flag
23. SYN	D	Syn TCP flag
24. FIN	D	Fin TCP flag
25. Land	D	1 if connection is from/to the same host/port; 0 otherwise
Content-based features		
26. Mss-src-dest-requested	C	Maximum segment size from source to destination requested
27. Mss-dest-src-requested	C	Maximum segment size from destination to source requested
28. Ttt-len-src-dst	C	Time to live length from source to destination
29. Ttt-len-dst-src	C	Time to live length from destination to source
30. Conn-status	C	Status of the connection (e.g., '1' for complete, '0' for reset)
Time-based features		
31. count-fr-dest	C	Number of frames received by unique destinations in the last T seconds from the same source
32. count-fr-src	C	Number of frames received from unique sources in the last T seconds from the same destination
33. count-serv-src	C	Number of frames from the source to the same destination port in the last T seconds
34. count-serv-dest	C	Number of frames from destination to the same source port in the last T seconds
35. num-pushed-src-dst	C	The number of pushed packets flowing from source to destination
36. num-pushed-dst-src	C	The number of pushed packets flowing from destination to source
37. num-SYN-FIN-src-dst	C	The number of SYN/FIN packets flowing from source to destination
38. num-SYN-FIN-dst-src	C	The number of SYN/FIN packets flowing from destination to source
39. num-FIN-src-dst	C	The number of FIN packets flowing from source to destination
40. num-FIN-dst-src	C	The number of FIN packets flowing from destination to source
Connection-based features		
41. count-dest-conn	C	Number of frames to unique destinations in the last N packets from the same source
42. count-src-conn	C	Number of frames from unique sources in the last N packets to the same destination
43. count-serv-srcconn	C	Number of frames from the source to the same destination port in the last N packets
44. count-serv-destconn	C	Number of frames from the destination to the same source port in the last N packets
45. num-packets-src-dst	C	The number of packets flowing from source to destination
46. num-packets-dst-src	C	The number of packets flowing from destination to source
47. num-acks-src-dst	C	The number of acknowledgement packets flowing from source to destination
48. num-acks-dst-src	C	The number of acknowledgement packets flowing from destination to source
49. num-retransmit-src-dst	C	The number of retransmitted packets flowing from source to destination
50. num-retransmit-dst-src	C	The number of retransmitted packets flowing from destination to source
C-Continuous, D-Discrete		

datasets will help to identify individual attacks in more effectively in high speed networks.

4 Observations and Conclusion

Several questions may be raised with respect to what constitutes a perfect dataset when dealing with the datasets generation task. These include qualities of normal, anomalous or realistic traffic included in the dataset. We provide a path and a template to generate a dataset that simultaneously exhibits the appropriate levels of normality, anomalousness and realism while avoiding the various weak points of currently available datasets, pointed out earlier. Quantitative measurements can be obtained only when specific methods are applied to the dataset.

The following are the major observations and requirements when generating an unbiased real-life dataset for intrusion detection.

- The dataset should not exhibit any unintended property in both normal and anomalous traffic.
- The dataset should be labelled properly.
- The dataset should cover all possible current network scenarios.
- The dataset should be entirely nonanonymized.
- In most benchmark datasets, the two basic assumptions described in Section 1 are valid but this bias should be avoided as much as possible.

Table 12: List of flow level features in TUIDS intrusion dataset

Label/feature name	*Type*	*Description*
Basic features		
1. Duration	C	Length (number of seconds) of the flow
2. Protocol-type	D	Type of protocol, e.g., TCP, UDP, ICMP
3. Src-ip	C	Source host IP address
4. Dest-ip	C	Destination IP address
5. Src-port	C	Source host port number
6. Dest-port	C	Destination host port number
7. ToS	D	Type of service
8. URG	D	TCP urgent flag
9. ACK	D	TCP acknowledgement flag
10. PSH	D	TCP push flag
11. RST	D	TCP reset flag
12. SYN	D	TCP SYN flag
13. FIN	D	TCP FIN flag
14. Src-bytes	C	Number of data bytes transfered from source to destination
15. Dest-bytes	C	Number of data bytes transfered from destination to source
16. Land	D	1 if connection is from/to the same host/port; 0 otherwise
Content-based features		
17. Conn-status	C	Status of the connection (e.g., '1' for complete, '0' for reset)
Time-based features		
18. count-dest	C	Number of flows to unique destination IPs in the last T seconds from the same source
19. count-src	C	Number of flows from unique source IPs in the last T seconds to the same destination
20. count-serv-src	C	Number of flows from the source to the same destination port in the last T seconds
21. count-serv-dest	C	Number of flows from the destination to the same source port in the last T seconds
Connection-based features		
22. count-dest-conn	C	Number of flows to unique destination IPs in the last N flows from the same source
23. count-src-conn	C	Number of flows from unique source IPs in the last N flows to the same destination
24. count-serv-srcconn	C	Number of flows from the source IP to the same destination port in the last N flows
25. count-serv-destconn	C	Number of flows to the destination IP to the same source port in the last N flows
C-Continuous, D-Discrete		

- Several datasets lack traffic features, although it is important to extract traffic features with their relevancy for a particular attack.

Despite the effort needed to create unbiased datasets, there will always be deficiencies in any one particular dataset. Therefore, it is very important to generate dynamic datasets which not only reflect the traffic compositions and intrusions types of the time, but are also modifiable, extensible, and reproducible. Therefore, new datasets must be generated from time to time for the purpose of analysis, testing and evaluation of network intrusion detection methods and systems from multiple perspectives.

In this paper, we provide a systematic approach to generate real-life network intrusion datasets using both packet and flow level traffic information. Three different types of datasets has been generated using the TUIDS testbed. They are (i) the TUIDS intrusion dataset, (ii) the TUIDS coordinated scan dataset, and (iii) the TUIDS DDoS dataset. We incorporate the maximum number of possible attacks and scenarios when generating the datasets on our testbed network.

Acknowledgments

This work is partially supported by Department of Information Technology (DIT) and Council of Scientific & Industrial Research (CSIR), Government of India. The authors are thankful to the funding agencies and also gratefully acknowledge the anonymous reviewers for their valuable comments.

References

[1] M. H. Bhuyan, D. K. Bhattacharyya, and J. K. Kalita, "RODD: An effective reference-based outlier detection technique for large datasets," in *Proceedings of First International Conference on Computer Science and Information Technology*, pp. 76–84, Bangalore, India, 2011.

[2] M. H. Bhuyan, D. K. Bhattacharyya, and J. K. Kalita, "Network anomaly detection: Methods, systems and tools," *IEEE Communications Surveys and Tutorials*, vol. 16, no. 1, pp. 303–336, 2014.

[3] M. H. Bhuyan, D. K. Bhattacharyya, and J. K. Kalita, "Towards an unsupervised method for network anomaly detection in large datasets," *Computing and Informatics*, vol. 33, no. 1, pp. 1–34, 2014.

[4] M. H. Bhuyan, D. K. Bhattacharyya, and J. K. Kalita, "NADO: Network anomaly detection using outlier approach," in *Proceedings of ACM International Conference on Communication, Computing & Security*, pp. 531–536, New York, USA, 2011.

[5] M. H. Bhuyan, D. K. Bhattacharyya, and J. K. Kalita, "AOCD: An adaptive outlier based coordinated scan detection approach," *International Journal of Network Security*, vol. 14, no. 6, pp. 339–351, 2012.

[6] M. H. Bhuyan, D. K. Bhattacharyya, and J. K. Kalita, "An effective unsupervised network anomaly detection method," in *Proceedings of ACM International Conference on Advances in Computing, Communications and Informatics*, pp. 533–539, New York, USA, 2012.

[7] CACE Technologies, *WinPcap*, June 2015. (http://www.winpcap.org)

[8] CAIDA, *The Cooperative Analysis for Internet Data Analysis*, 2011. (http://www.caida.org)

Table 13: List of features in the KDDcup99 intrusion dataset

Label/feature name	Type	Description
Basic features		
1. Duration	C	Length (number of seconds) of the connection
2. Protocol-type	D	Type of protocol, e.g., tcp, udp, etc.
3. Service	D	Network service at the destination, e.g., http, telnet, etc.
4. Flag	D	Normal or error status of the connection
5. Src-bytes	C	Number of data bytes from source to destination
6. Dst-bytes	C	Number of data bytes from destination to source
7. Land	D	1 if connection is from/to the same host/port; 0 otherwise
8. Wrong-fragment	C	Number of "wrong" fragments
9. Urgen	C	Number of urgent packets
Content-based features		
10. Hot	C	Number of "hot" indicators (hot: number of directory accesses, create and execute program)
11. Num-failed-logins	C	Number of failed login attempts
12. Logged-in	D	1 if successfully logged-in; 0 otherwise
13. Num-compromised	C	Number of "compromised" conditions (compromised condition: number of file/path not found errors and jumping commands)
14. Root-shell	D	1 if root-shell is obtained; 0 otherwise
15. Su-attempted	D	1 if "su root" command attempted; 0 otherwise
16. Num-root	C	Number of "root" accesses
17. Num-file-creations	C	Number of file creation operations
18. Num-shells	C	Number of shell prompts
19. Num-access-files	C	Number of operations on access control files
20. Num-outbound-cmds	C	Number of outbound commands in an ftp session
21. Is-host-login	D	1 if login belongs to the "hot" list; 0 otherwise
22. Is-guest-login	D	1 if the login is a "guest" login; 0 otherwise
Time-based features		
23. Count	C	Number of connections to the same host as the current connection in the past 2 seconds
24. Srv-count	C	Number of connections to the same service as the current connection in the past 2 seconds (same-host connections)
25. Serror-rate	C	% of connections that have "SYN" errors (same-host connections)
26. Srv-serror-rate	C	% of connections that have "SYN" errors (same-service connections)
27. Rerror-rate	C	% of connections that have "REJ" errors (same-host connections)
28. Srv-rerror-rate	C	% of connections that have "REJ" errors (same-service connections)
29. Same-srv-rate	C	% of connections to the same service (same-host connections)
30. Diff-srv-rate	C	% of connections to different services (same-host connections)
31. Srv-diff-host-rate	C	% of connections to different hosts (same-service connections)
Connection-based features		
32. Dst-host-count	C	Count of destination hosts
33. Dst-host-srv-count	C	Srv_count for destination host
34. Dst-host-same-srv-rate	C	Same_srv_rate for destination host
35. Dst-host-diff-srv-rate	C	Diff_srv_rate for destination host
36. Dst-host-same-src-port-rate	C	Same_src_port_rate for destination host
37. Dst-host-srv-diff-host-rate	C	Diff_host_rate for destination host
38. Dst-host-serror-rate	C	Serror_rate for destination host
39. Dst-host-srv-serror-rate	C	Srv_serror_rate for destination host
40. Dst-host-rerror-rate	C	Rerror_rate for destination host
41. Dst-host-srv-rerror-rate	C	Srv_rerror_rate for destination host
C-Continuous, D-Discrete		

Table 14: TUIDS dataset traffic composition

Protocol	*Size (MB)*	*(%)*
(a) Total traffic composition		
IP	66784.29	99.99
ARP	3.96	0.005
IPv6	0.00	0.00
IPX	0.00	0.00
STP	0.00	0.00
Other	0.00	0.00
(b) TCP/UDP/ICMP traffic composition		
TCP	49049.29	73.44%
UDP	14940.53	22.37%
ICMP	2798.43	4.19%
ICMPv6	0.00	0.00
Other	0.00	0.00

[9] A. Cemerlic, L. Yang, and J.M. Kizza, "Network intrusion detection based on bayesian networks," in *Proceedings of the 20th International Conference on Software Engineering and Knowledge Engineering*, pp. 791–794, San Francisco, USA, 2008.

[10] A. Dainotti and A. Pescape, "PLAB: A packet capture and analysis architecture," 2004. (http://traffic.comics.unina.it/software/ITG/D-ITGpublications/TR-DIS-122004.pdf)

[11] DEFCON, *The SHMOO Group*, 2011. (http://cctf.shmoo.com/)

[12] L. Delooze, *Applying Soft-Computing Techniques to Intrusion Detection*, Ph.D. Thesis, Computer Science Department, University of Colorado, Colorado Springs, 2005.

[13] D. E. Denning, "An intrusion-detection model," *IEEE Transactions on Software Engineering*, vol. 13, pp. 222–232, Feb. 1987.

[14] A. A. Ghorbani, W. Lu, and M. Tavallaee, "Network attacks," in *Network Intrusion Detection and Prevention*, pp. 1–25, Springer-verlag, 2010.

[15] P. Gogoi, D. K. Bhattacharyya, B. Bora, and J. K. Kalita, "MLH-IDS: A multi-level hybrid intrusion detection method," *The Computer Journal*, vol. 57, pp. 602–623, May 2014.

[16] P. Gogoi, M. H. Bhuyan, D. K. Bhattacharyya, and J. K. Kalita, "Packet and flow-based network intrusion dataset," in *Proceedings of the 5th International Conference on Contemporary Computing*, LNCS-CCIS 306, pp. 322–334, Springer, 2012.

Table 15: Distribution of normal and attack connection instances in real time packet and flow level TUIDS datasets

Connection type	Dataset type			
	Training dataset		Testing dataset	
(a) TUIDS intrusion dataset				
Packet level				
Normal	71785	58.87%	47895	55.52%
DoS	42592	34.93%	30613	35.49%
Probe	7550	6.19%	7757	8.99%
Total	121927	-	86265	-
Flow level				
Normal	23120	43.75%	16770	41.17%
DoS	21441	40.57%	14475	35.54%
Probe	8282	15.67%	9480	23.28%
Total	52843	-	40725	-
(b) TUIDS coordinated scan dataset				
Packet level				
Normal	65285	90.14%	41095	84.95%
Probe	7140	9.86%	7283	15.05%
Total	72425	-	48378	-
Flow level				
Normal	20180	73.44%	15853	65.52%
Probe	7297	26.56%	8357	34.52%
Total	27477	-	24210	-
(c) TUIDS DDoS dataset				
Packet level				
Normal	46513	68.62%	44328	60.50%
Flooding attacks	21273	31.38%	28936	39.49%
Total	67786	-	73264	-
Flow level				
Normal	27411	57.67%	28841	61.38%
Flooding attacks	20117	42.33%	18150	38.62%
Total	47528	-	46991	-

Table 16: Comparison of existing datasets and their characteristics

Dataset	*u*	*v*	*w*	*No. of instances*	*No. of attributes*	*x*	*y*	*z*	*Some references*
Synthetic	No	No	Yes	user dependent	user dependent	Not known	any	user dependent	[4, 1]
KDDcup99	Yes	No	Yes	805050	41	BCTW	P	C_1	[48, 33, 47, 31]
NSL-KDD	Yes	No	Yes	148517	41	BCTW	P	C_1	[43]
DARPA 2000	Yes	No	No	Huge	Not known	Raw	Raw	C_2	[37]
DEFCON	No	No	No	Huge	Not known	Raw	P	C_2	[37]
CAIDA	Yes	Yes	No	Huge	Not known	Raw	P	C_1	[37]
LBNL	Yes	Yes	No	Huge	Not known	Raw	P	C_2	[46]
Endpoint	Yes	Yes	No	Huge	Not known	Raw	P	C_2, C_3	[46]
UNIBS	Yes	Yes	No	Huge	Not known	Raw	P	C_2	[46]
ISCX-UNB	Yes	Yes	Yes	Huge	Not known	Raw	P	A	[37]
KU	Yes	Yes	No	Huge	24	BTW	P	C_1	[39]
TUIDS	Yes	Yes	Yes	Huge	50,24	BCTW	P,F	C_1	[4, 1]

u-realistic network configuration
v-indicates realistic traffic
w-describes the label information
x-types of features extracted as basic features (B), content based features (C), time based features(T) and window based features(W)
y-explains the types of data as packet based (P) or flow based (F) or hybrid (H) or others (O)
z-represents the attack category as C_1-all attacks, C_2-denial of service, C_3-probe, C_4-user to root, C_5-remote to local, and A-application layer attacks

[17] D. Hoplaros, Z Tari, and I. Khalil, "Data summarization for network traffic monitoring," *Journal of Network and Computer Applications*, vol. 37, pp. 194–205, 2014.

[18] Information Systems Technology Group MIT Lincoln Lab, *DARPA Intrusion Detection Data Sets*, Mar. 2000. (http://www.ll.mit.edu/mission/ communications/ist/corpora/ideval/data/2000data.html)

[19] V. Jacobson, C. Leres, and S. McCanne, "The tcpdump manual page," Lawrence Berkeley Laboratory, Berkeley, CA, 1989.

[20] V. Jacobson, C. Leres, and S. McCanne, "Libpcap," Lawrence Berkeley Laboratory, Berkeley, CA, Initial public release, June 1994.

[21] KDDcup99, "Knowledge discovery in databases DARPA archive," 1999. (https://archive.ics.uci.edu/ml/databases/kddcup99/)

[22] K. Kendall, *A Database of Computer Attacks for the Evaluation of Intrusion Detection Systems*, Master's Thesis, MIT, 1999.

[23] Lawrence Berkeley National Laboratory (LBNL), *ICSI, LBNL/ICSI Enterprise Tracing Project*, 2005. (http://www.icir.org/enterprise-tracing/)

[24] A. Lazarevic, L. Ertoz, V. Kumar, A. Ozgur, and J. Srivastava, "A comparative study of anomaly detection schemes in network intrusion detection," in *Proceedings of the 3rd SIAM International Conference on Data Mining*, pp. 25–36, 2003.

[25] B. Li, J. Springer, G. Bebis, and M. H. Gunes, "A survey of network flow applications," *Journal of Network*

and Computer Applications, vol. 36, no. 2, pp. 567–581, 2013.

[26] R. P. Lippmann, D. J. Fried, I. Graf, et al., "Evaluating intrusion detection systems: The 1998 DARPA offline intrusion detection evaluation," in *Proceedings of the DARPA Information Survivability Conference and Exposition*, pp. 12–26, 2000.

[27] M. V. Mahoney and P. K. Chan, "An analysis of the 1999 DARPA/Lincoln laboratory evaluation data for network anomaly detection," in *Proceedings of the 6th International Symposium on Recent Advances in Intrusion Detection*, pp. 220–237, 2003.

[28] S. McCanne and V. Jacobson, "The BSD packet filter: A new architecture for user level packet capture," in *Proceedings of the Winter 1993 USENIX Conference*, pp. 259–269, 1993.

[29] J. McHugh, "Testing intrusion detection systems: a critique of the 1998 and 1999 DARPA intrusion detection system evaluations as performed by lincoln laboratory," *ACM Transactions on Information and System Security*, vol. 3, pp. 262–294, Nov. 2000.

[30] P. Mell, V. Hu, R. Lippmann, J. Haines, and M. Zissman, *An Overview of Issues in Testing Intrusion Detection Systems*, 2003. (http://citeseer.ist.psu.edu/621355.html)

[31] Z. Muda, W. Yassin, M. N. Sulaiman, and N. I. Udzir, "A K-means and naive bayes learning approach for better intrusion detection," *Information Technology Journal*, vol. 10, no. 3, pp. 648–655, 2011.

[32] NSL-KDD, *NSL-KDD Data Set for Network-based Intrusion Detection Systems*, Mar. 2009. (http://iscx.cs.unb.ca/NSL-KDD/)

[33] M. E. Otey, A. Ghoting, and S. Parthasarathy, "Fast distributed outlier detection in mixed-attribute data sets," *Data Mining and Knowledge Discovery*, vol. 12, no. 2-3, pp. 203–228, 2006.

[34] R. Pang, M. Allman, M. Bennett, J. Lee, V. Paxson, and B. Tierney, "A first look at modern enterprise traffic," in *Proceedings of the 5th ACM SIGCOMM Conference on Internet Measurement*, pp. 2, Berkeley, USA, 2005.

[35] R. Pang, M. Allman, V. Paxson, and J. Lee, "The devil and packet trace anonymization," *SIGCOMM Computer Communication Review*, vol. 36, no. 1, pp. 29–38, 2006.

[36] L. Portnoy, E. Eskin, and S. Stolfo, "Intrusion detection with unlabeled data using clustering," in *Proceedings of ACM CSS Workshop on Data Mining Applied to Security*, pp. 5–8, 2001.

[37] A. Shiravi, H. Shiravi, M. Tavallaee, and A. A. Ghorbani, "Towards developing a systematic approach to generate benchmark datasets for intrusion detection," *Computers & Security*, vol. 31, no. 3, pp. 357–374, 2012.

[38] J. Song, H. Takakura, and Y. Okabe, "Description of kyoto university benchmark data,". pp. 1–3. 2006. (http://www.takakura.com/Kyoto_data/BenchmarkData-Description-v5.pdf)

[39] J. Song, H. Takakura, Y. Okabe, and K. Nakao, "Toward a more practical unsupervised anomaly detection system," *Information Sciences*, vol. 231, pp. 4–14, Aug. 2013.

[40] A. Sperotto, R. Sadre, F. Vliet, and A. Pras, "A labeled data set for flow-based intrusion detection," in *Proceedings of the 9th IEEE International Workshop on IP Operations and Management*, pp. 39–50, Venice, Italy, 2009.

[41] S. J. Stolfo, W. Fan, W. Lee, A. Prodromidis, and P. K. Chan, "Cost-based modeling for fraud and intrusion detection: Results from the JAM project," in *Proceedings of the IEEE DARPA Information Survivability Conference and Exposition*, vol. 2, pp. 130–144, USA, 2000.

[42] symantec.com, *Symantec Security Response*, June 2015. (http://securityresponse.symantec.com/avcenter)

[43] M. Tavallaee, E. Bagheri, W. Lu, and A. A. Ghorbani, "A detailed analysis of the KDD CUP 99 data set," in *Proceedings of the 2nd IEEE International Conference on Computational Intelligence for Security and Defense Applications*, pp. 53–58, USA, 2009.

[44] C. Thomas, V. Sharma, and N. Balakrishnan, "Usefulness of DARPA dataset for intrusion detection system evaluation," in *Proceedings of the Data Mining, Intrusion Detection, Information Assurance, and Data Networks Security*, SPIE 6973, Orlando, FL, 2008.

[45] UNIBS, *University of Brescia Dataset*, 2009. (http://www.ing.unibs.it/ntw/tools/traces/)

[46] J. Xu and C. R. Shelton, "Intrusion detection using continuous time bayesian networks," *Journal of Artificial Intelligence Research*, vol. 39, pp. 745–774, 2010.

[47] G. Zhang, S. Jiang, G. Wei, and Q. Guan, "A prediction-based detection algorithm against distributed denial-of-service attacks," in *Proceedings of the ACM International Conference on Wireless Communications and Mobile Computing: Connecting the World Wirelessly*, pp. 106–110, Leipzig, Germany, 2009.

[48] Y. F. Zhang, Z. Y. Xiong, and X. Q. Wang, "Distributed intrusion detection based on clustering," in *Proceeding of the International Conference on Machine Learning and Cybernetics*, vol. 4, pp. 2379–2383, Aug. 2005.

Monowar H. Bhuyan is an assistant professor in the Department of Computer Science and Engineering at Kaziranga University, Jorhat, Assam, India. He received his Ph.D. in Computer Science & Engineering from Tezpur University (a Central University) in February 2014. He is a life member of IETE, India. His research areas include data mining, cloud security, computer and network security. He has published 20 papers in international journals and referred conference proceedings. He is on the programme committee members/referees of several international conferences/journals.

Dhruba K. Bhattacharyya received his Ph.D. in Computer Science from Tezpur University in 1999. Currently, he is a Professor in the Computer Science & Engineering Department at Tezpur University. His research areas include data mining, network security and bioinformatics. Prof. Bhattacharyya has published more than 220 research papers in leading international journals and conference proceedings. Dr. Bhattacharyya also has written/edited 10 books. He is on the editorial boards of several international journals and also on the programme committees/advisory bodies of several international conferences/workshops.

Jugal K. Kalita is a professor of Computer Science at the University of Colorado at Colorado Springs. He received his Ph.D. from the University of Pennsylvania in 1990. His research interests are in natural language processing, machine learning, artificial intelligence, bioinformatics and applications of AI techniques to computer and network security. He has published more than 150 papers in international journals and referred conference proceedings and has written two technical books. Professor Kalita is a frequent visitor of Tezpur University where he collaborates on research projects with faculty and students.

An Efficient and Practical Authenticated Communication Scheme for Vehicular Ad Hoc Networks

Chin-Chen Chang[1], Jen-Ho Yang[2], and Yu-Ching Wu[3]
(Corresponding author: Jen-Ho Yang)

Department of Information Engineering and Computer Science, Feng Chia University[1]
100 Wenhwa Rd., Seatwen, Taichung 40724, Taiwan, R.O.C.
Department of Information and Electronic Commerce, Kainan University[2]
No. 1, Kannan Rd., Luzhu, Taoyuan County, 33857, Taiwan, R.O.C.
(Email: jenhoyang@mail.knu.edu.tw)
Department of Computer Science and Information Engineering, National Chung Cheng University[3]
160 San-Hsing, Ming-Hsiung, Chiayi 621, Taiwan, R.O.C.
(Received Mar. 4, 2013; revised and accepted May 16 & Aug. 13, 2013)

Abstract

In the vehicular ad hoc networks (VANET), various authentication schemes have been proposed for secure communications. However, the previous schemes are inefficient because each vehicle needs to share and keep a large number of session keys for communicating with the other vehicles on the VANET. To overcome the above drawback, we propose a new authenticated communication scheme for the VANET. In the proposed scheme, each vehicle communicates with the other vehicles through the roadside unit (*RSU*). Based upon this environment, each vehicle only has to share a session key with the *RSU* to communicate with different vehicles. Thus, the proposed communication model can be simplified on the VANET.

Keywords: Authentication, elliptic curve cryptography, mobile ad hoc networks, vehicular ad hoc networks, wireless communication

1 Introduction

Mobile ad hoc networks (MANET) [1, 3, 14, 17, 19, 21] is a network architecture which combines ad hoc and wireless networks. The MANET does not require a fixed network infrastructure to keep the network connection and it is self-organized. In the MANET applications, a vehicular ad hoc network (VANET) is the most popular one because it provides a secure environment for vehicular communications. However, some characteristics of the VANET are different from those of the MANET. For example, the vehicle speed on the VANET is faster than the mobile node in the MANET, and the network topology of the VANET is deployed according to the direction of roadway. The main goal of the VANET is to provide the driving safety and comfortable to users. The applications of the VANET can be divided into two parts: the Intelligent Transportation system (ITS) application and the comfortable application [5, 11, 13, 16].

Generally, the ITS is used to provide the transmission safety of vehicle communications and increase the driving efficiency. The ITS applications include the control of traffic flows, preventing the car collisions, analyzing the traffic jams, evaluating the traffic situations, and deciding the driving routes and so on. For example, a vehicle can broadcast the accident message to caution the other incoming vehicles while a vehicle accident happens. Then, the incoming vehicles can select other driving routes to prevent this traffic jam so the possibility of the traffic accident can be reduced.

Besides, the comfortable application on the VANET is to provide the network connections for vehicles so the passengers in vehicles can derive some electronic services. For example, the passenger can easily download the electronic music, games, and E-mails in a vehicle.

From the business or commercial point of view, the VANET has the commercial potential for many applications so it becomes a popular research in recent years. For the communication security, many secure communication schemes for the VANET have been proposed [5, 12, 13, 16]. For traffic control on the VANET, Li et al. [11] proposed a secure model with three communication schemes based on ID-based cryptography [10, 15, 20] and the blind digital signature schemes [2, 4, 18]. In addition, they also proposed an entertainment service scheme with privacy preservation for the VANET.

However, we found that Li et al.'s communication

model for traffic control is too complicated and inefficient. This is because that each vehicle needs to share and keep a large number of session keys for communicating with the other vehicles in their scheme. Moreover, Li et al.'s communication model is impractical because a vehicle needs to perform different communication schemes to communicate with different roles on the VANET. Besides, their entertainment service scheme is also inefficient and impractical because it has unnecessary communications between the vehicle and the service provider.

To overcome the above-mentioned drawbacks, we propose an efficient authenticated communication scheme for the VANET. In the proposed scheme, a vehicle communicates with the other vehicles through the roadside unit (*RSU*), which is set on the roadside to broadcast and receive messages for vehicles. Based upon this environment, a vehicle only needs to share a session key with the *RSU* to communicate with a large number of vehicles. Besides, the proposed scheme integrates Li et al.'s three communication schemes so the infrastructure of the proposed scheme is more practical and simpler for the VANET.

Besides, we also propose an entertainment service scheme for the VANET without involving the service provider. In the proposed service scheme, the function of the service provider is integrated into the *RSU*. Therefore, the communication and computation costs can be drastically reduced when the passenger requires the entertainment services in a vehicle. According to the above-mentioned advantages, the proposed scheme is more efficient and practical than the previously proposed schemes for the VANET.

2 The Related Work

In this section, we briefly describe Li et al.'s scheme [11] and its drawbacks.

2.1 Li et al.'s Scheme

There are three roles in Li et al.'s scheme: the vehicle, the roadside unit (*RSU*), and the service provider. In this system, each vehicle is equipped with a mobile device to communicate with the other vehicles and the *RSU*. The *RSU* is responsible for broadcasting traffic information or entertainment applications to the vehicles. And, the service provider is responsible for providing some entertainment services to passengers in a vehicle. In [11], Li et al. proposed three communication models for the VANET: the vehicle-to-vehicle communication, the vehicle-to-RSU communication, and the RSU-to-vehicle communication models. Besides, Li et al. also proposed a secure and efficient communication scheme with privacy preservation (SECSPP) for entertainment applications on the VANET. The notations used in Li et al.'s schemes are shown in Table 1. Now, we describe Li et al.'s schemes as follows.

2.1.1 The Vehicle-to-Vehicle Communication Scheme

Assume that a vehicle V_i wants to communication with another vehicle V_j, the detailed steps are shown as follows.

1) V_i selects a random number a and $tag\#$. Next, V_i computes $M = C \oplus (tag\#||ID_{V_i}||ID_{V_j}||T_{V_i}||a)$ and $C = (ID_{V_i}^2)^{H(T_{V_i}||r)K_{V_i}}$, where T_{V_i} is a timestamp, r is the roadway section, and K_{V_i} is the secret key of V_i.

2) V_i broadcasts $H'(SK) \oplus (tag\#,\ ID_{V_i},\ ID_{V_j},\ hop,\ r,\ T_{V_j},\ M)$ to the vehicles within V_i's transmission range, where $H'(SK)$ is the shared secret key in the network.

3) After receiving $H'(SK) \oplus (tag\#, ID_{V_i}, ID_{V_j}, hop, r, T_{V_j}, M)$, V_j decrypts the message by $H'(SK)$. Then, V_j computes $C' = (ID_{V_i}^2)^{H(T_{V_i}||r)K_{V_i}}$ to reveal S. And, V_j checks the validity of hop and ID_{V_j}. If they are valid, then V_j selects a random number b to compute a session key $K_{V_j,V_i} = H(a||b||0)$.

4) V_j sends $H'(SK) \oplus (tag\#, ID_{V_i}, ID_{V_j}, T_{V_j}, r, S')$ to V_i, where $M' = C' \oplus (tag\#||ID_{V_i}\ ||ID_S\ ||T_{V_i}||r||b||MAC)$ and $MAC = H(K_{V_j,V_i}; a+1)$.

5) After receiving $H'(SK) \oplus (tag\#,\ ID_{V_i},\ ID_{V_j},\ T_{V_j},\ r, S')$, A reveals $(tag\#, ID_{V_i}, ID_{V_j}, T_{V_j}, r||b||MAC)$. Then, V_i can compute the session key $K_{V_i,V_j} = H(a||b||0)$ and verifies the correctness of MAC. If the above verifications hold, then V_i and V_j can share a common session key and use it to communicate with each other.

2.1.2 The Vehicle-to-RSU Communication Scheme

Assume that an ambulance V_A transmits an emergency signal to the *RSU*, then V_A can control traffic lights on its way to a hospital. The detailed steps are shown as follows.

1) V_A generates a random number a to compute $M = C \oplus (ES||ID_{V_A}\ ||ID_R\ ||T_{V_i}||a)$ and $C = (ID_R^2)^{H(T_{V_A}||r)K_{V_A}}$, where ID_R is the identity of *RSU*, ES is the emergency signal, and K_A is the secret key. Then, V_A sends $H'(SK) \oplus (ES,\ ID_{V_A},\ ID_R,\ r,\ T_{V_A},\ M)$ to R.

2) Upon receiving the above messages, R reveals the message by $H'(SK)$ and checks the validity of V_A. If the above verification is correct, then *RSU* computes $C' = (ID_{V_A}^2)^{H(T_{V_A}||r)K_R}$ to reveal S. Afterward, *RSU* selects a random number b to compute the session key $K_{R,V_A}(a||b||0)$.

3) *RSU* sends $H'(SK) \oplus (ES,\ ID_R,\ ID_{V_A},\ r,\ T_R,\ S')$ to A, where $M' = C' \oplus (ES||ID_R||ID_{V_A}\ ||T_R||r||b||MAC)$ and $MAC = H(K_{R,V_A}||a+1)$.

Table 1: The notations of Li et al.'s scheme

ID_X	The identity of the entity X
$PK'_S SK_S$	The public/private key of the service provider
K_X	The secret key of the entity X
$tag\#$	An unique tag number for a request
hop	The number of hops
r	The identity of roadway section
ES	An emergency signal
MAC	The message authentication code
$H(\cdot)$	A collision-free and public one-way hash function
M_X	The receipt of the service access for the vehicle X
T_X	A timestamp generated by the entity X
$H(SK)$	The group secret key shared among all nodes in the VANET
$\|\|$	The concatenation operation
$E_{PK_S}\{\cdot\}$	The asymmetric encryption function using the public key
$D_{SK_S}\{\cdot\}$	The asymmetric decryption function using the private key

4) After receiving the above messages, V_A uses $H'(SK)$ and C to reveal $(ES||ID_R||ID_{V_A}||T_B||r||b||MAC)$. Next, V_A computes $K_{V_A,R} = (a||b||0)$ to verify the correctness of MAC. If the above verifications are correct, then V_A and RSU can use the session key to communicate with each other.

2.1.3 The RSU-to-Vehicle Communication Scheme

Assume that RSU wants to update the shared group key $H'(SK)$ to all vehicles within its transmission range, and then the detailed steps are shown as follows.

1) RSU generates a new shared key c and $nonce_R$. Next, the RSU broadcasts the following message $H'(SK) \oplus (Update\_Key,\ H^{t-1}(SK),\ ID_R,\ r,\ T_R,\ nonce_R)$ to all vehicles in its transmission range.

2) After V_i receiving the following message: $H'(SK) \oplus (Update\_Key, H^{t-1}(SK), ID_R, r, T_R, nonce_R)$, then V_i can decrypt it by using $H'(SK)$. Next, V_i verifies the shared key $H^{t-1}(SK)$ by checking if the equation $H'(SK) = H(H^{t-1}(SK))$ holds or not. If the equation holds, then V_i updates the shared key with $H^{t-1}(SK)$ and broadcasts the following message $H^{t-1}(SK) \oplus (ID_{V_i}, T_{V_i}, r, nonce_R + 1)$ to RSU.

3) After receiving $H^{t-1}(SK) \oplus (ID_{V_i}, T_{V_i}, r, nonce_R + 1)$, RSU can obtain $(ID_{V_i}, T_{V_i}, r, nonce_R + 1)$ by $H^{t-1}(SK) \oplus (ID_{V_i}, T_{V_i}, r, nonce_R + 1) \oplus H^{t-1}(SK)$. Then, RSU verifies if $(nonce_R + 1)$ is correct or not. If it is correct, then RSU knows that V_i has updated its shared key.

2.2 The Drawbacks of Li et al.'s Scheme

For the traffic control, Li et al. proposed a communication model containing three schemes: the vehicle-to-vehicle, the vehicle-to-RSU, and the RSU-to-vehicle communication schemes. However, this model is too complicated and inefficient. For example, in Li et al.'s vehicle-to-vehicle scheme, a vehicle V_i needs to share a session key and keep it to communicate with another vehicle V_j. If V_i wants to communicate with a large amount of vehicles, then V_i also needs to share and keep a large number of session keys for different vehicles. To communicate with RSU, the vehicle V_i also needs to share another session key with RSU. This drawback increases the communication and computation costs of each vehicle in Li et al.'s communication model for the VANET. In addition, to communicate with another vehicle or RSU, each vehicle needs to perform three different schemes. This drawback also makes Li et al.'s model impractical for the VANET.

3 The Proposed Scheme

For the traffic control on the VANET, Li et al. proposed a model containing three communication schemes as follows: the vehicle-to-vehicle, the vehicle-to-RSU, and the RSU-to-vehicle communication schemes. However, we point out that this model is inefficient and impractical in Subsection 2.3. If a vehicle V_i can communicate with the other vehicles through RSU, then V_i only needs to share and keep one session key for RSU on the VANET. Based upon this conception, we propose an efficient vehicle-RSU-vehicle communication scheme for the VANET in this section. Then, Li et al.'s three communication schemes can be simply simplified by the proposed scheme. Therefore, the proposed communication model for the traffic control on the VANET is more efficient and simpler than Li et al.'s model.

Table 2 shows the notations used in the proposed schemes. Now, we present the proposed schemes as follows.

Before describing the proposed scheme, we define some

Table 2: The Notations of the proposed scheme

ID_X	The identity of the entity X
K_{V_i}	A pre-shared key shared between a vehicle V_i and RSU
M	The transmitted message such as traffic information and emergency signal
T_X	A timestamp generated by the entity X
$H(\cdot)$	A secure one-way hash function
ES	The entertainment service such as online music and movies
x	The secret key of RSU
$\|\|$	The concatenation operation

notations as follows. In the proposed scheme, the system chooses $E_p(a,b)$: $y^2 = x^3 + ax + b(\mathrm{mod}p)$ over a prime finite field F_p with the order n, where $a, b \in F_p$, $p > 3$, and $4a^3 + 27b^2 \neq 0(\mathrm{mod}p)$ [6, 7, 8, 9]. Then, the system selects $x \in Z_n*$ to be the secret key of RSU and computes $X = x*Q$ to be the public key of RSU, where Q is a base point over E_p and "*" is the point multiplication over E_p. When a vehicle V_i wants to join into the proposed system, V_i first registers with RSU. Then, RSU generates a pre-shared key $K_{V_i} = H(ID_{V_i}||x)$ for V_i, and thus V_i can use K_{V_i} to communicate with RSU.

Assume that V_i wants to send the message M to another vehicle V_j, then V_i broadcasts M and some authentication information to RSU. Then, RSU can authenticate the source and the validity of M using the pre-shared key K_{V_i}. Also, RSU generates a signature for M using ECDSA [7] and broadcasts it to V_j. Finally, V_j can verify the signature to authenticate the validity of M. The detailed steps are shown as follows.

1) V_i broadcasts $\{ID_{V_i}.ID_{V_j}, M, T_{V_i}, K_{V_i} \oplus H(M||T_{V_i})\}$ to all vehicles and the RSU within its transmission range.

2) After receiving the above message, V_j does not need to authenticate it immediately. V_j just stores this message into its database until it receives the authenticated message from RSU. If V_j does not receive the authenticated message in a pre-defined expiration time, then it discards this message.

3) After receiving $\{ID_{V_i}, ID_{V_j}, M, T_{V_i}, K_{V_i} \oplus H(M||T_{V_i})\}$, RSU computes $K'_{V_i} = H(ID_{V_i}||x)$ according to ID_{V_i}. RSU computes $H'(M||T_{V_i}) = K'_{V_i} \oplus K_{V_i} \oplus H(M||T_{V_i})$. Then, RSU checks if the equation $H'(M||T_{V_i}) = H(M||T_{V_i})$ holds. If it holds, then RSU authenticates the validity of M and T_{V_i}.

4) RSU randomly selects an integer $t \in Z_n^*$ and computes $T = t * Q = (x_1, y_1)$, where x_1 and y_1 are x-coordinate and y-coordinate of T, respectively. RSU computes $r = x_1 \bmod n$ and $s = t^{-1} \cdot [H(M||T_R) + x \cdot t] \bmod n$. Finally, RSU broadcasts $\{ID_R, ID_{V_j}, M, (r,s), T_R\}$ to all vehicles within its transmission range.

5) After receiving the above authenticated message, V_j checks whether the received message is in its database or not. If the message exists, then V_j computes the following $H(M||T_R) \cdot s^{-1} \bmod n$, $r \cdot s^{-1} \bmod n$, and $(H(M||T_R) \cdot s^{-1}) * Q + (r \cdot s^{-1}) * X = (x'_1, y'_1)$. Then, RSU computes $r' = x'_1 \bmod n$ and checks if $r' = r$ holds. If it holds, then V_j confirms that the message is really sent from V_i and M is valid.

Figure 1 illustrates the steps of the proposed vehicle-RSU-vehicle communication scheme. To broadcast a large number of vehicles and RSU, a vehicle only needs to share and keep one session key with RSU in our scheme. Therefore, the proposed scheme greatly reduces the communication loads and computation costs.

Based on the proposed scheme, if RSU wants to broadcast a message M to a vehicle V_i, we only need to perform the similar steps according to Step 1 and Step 3. For example, the RSU replaces $\{ID_{V_i}, ID_{V_j}, M, T_{V_i}, K_{V_i} \oplus H(M||T_{V_i})\}$ with $\{ID_R, ID_{V_j}, M, T_R, K_{V_i} \oplus H(M||T_R)\}$ in Step 1, and then RSU broadcasts it in its broadcast range. Then, V_i can authenticate M by K_{V_i} according to the verification equations in Step 3. Note that only the correct V_i can verify the validity of M. Similarly, if V_i wants to broadcast a message to RSU, then V_i only needs to perform Step 1 and Step 3 by replacing some messages. Therefore, we successfully simplify Li et al.'s three schemes into the proposed vehicle-RSU-vehicle communication scheme.

4 The Security Analysis

To analyze the security of the two proposed schemes, we discuss some possible attacks as follows.

Replay attack. Assume that an attacker wiretaps the communications between the vehicles in the proposed vehicle-RSU-vehicle scheme, then the attacker can obtain $\{ID_{V_i}, ID_{V_j}, M, T_{V_i}, K_{V_i} \oplus H(M||T_{V_i})\}$. Furthermore, the attacker wants to re-broadcast the following message $\{ID_{V_i}.ID_{V_j}, M, T'_{V_i}, K_{V_i} \oplus H(M||T_{V_i})\}$ at the time T'_{V_i}. However, this attack cannot work because RSU computes K_{V_i} and checks if $H(M||T'_{V_i})$ is equal to $H(M||T_{V_i})$. Then, RSU can discover that the message $\{ID_{V_i}, ID_{V_j}, M, T'_{V_i}, K_{V_i} \oplus$

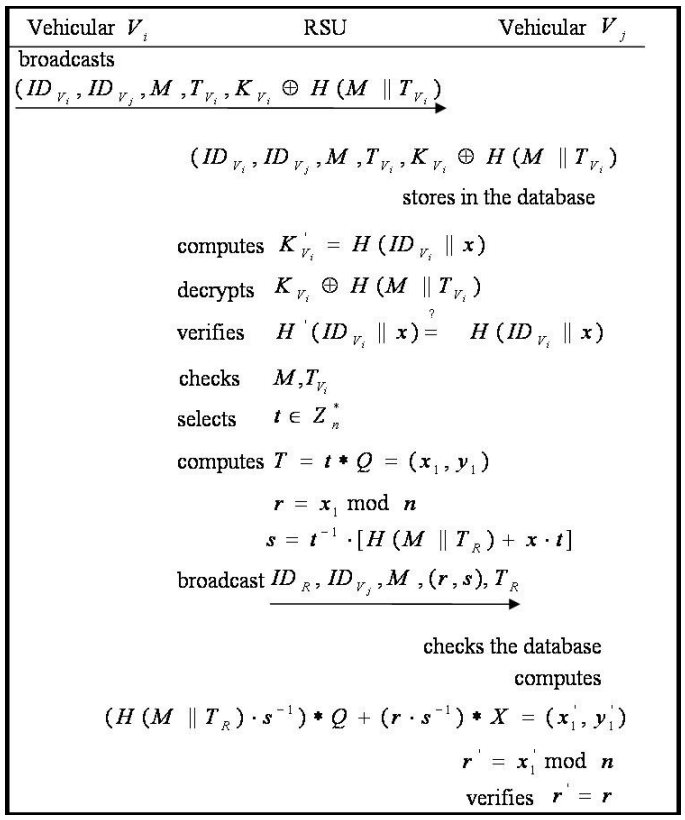

Figure 1: The proposed scheme

$H(M||T_{V_i})\}$ is sent by the attacker because of $H(M||T'_{V_i}) \neq H(M||T_{V_i})$. Hence, this attack is infeasible for the proposed scheme.

Impersonation attack. Assume that an attacker wants to impersonate the vehicle V_i to broadcast the following message $\{ID_{V_i}, ID_{V_j}, M^*, T^*_{V_i}, K^*_{V_i} \oplus H(M^*||T^*_{V_i})\}$ in the proposed vehicle-RSU-vehicle scheme, then he/she selects a random number $x* \in Z_n*$ to compute the pre-shared key $K^*_{V_i} = H(ID_{V_i}||x*)$. In addition, the attacker broadcasts $\{ID_{V_i}, ID_{V_j}, M^*, T^*_{V_i}, K^*_{V_i} \oplus H(M^*||T^*_{V_i})\}$. After receiving the message, RSU computes $K_{V_i} = H(ID_{V_i}||x)$ and checks if $H(M^*||T^*_{V_i})$ is equal to $K_{V_i} \oplus K^*_{V_i} \oplus H(M^*||T^*_{V_i})$ or not. Obviously, RSU can discover that the message $\{ID_{V_i}, ID_{V_j}, M^*, T^*_{V_i}, K^*_{V_i} \oplus H(M^*||T^*_{V_i})\}$ is broadcasted by the attacker because $K^*_{V_i} \neq K_{V_i}$. Therefore, this attack is impossible for the vehicle-RSU-vehicle scheme.

Outsider attack. Assume that an attacker wants to obtain the symmetric key K_{V_i}, then he/she intercepts the communication between a vehicle V_i and RSU to get the messages $\{ID_{V_i}, ID_{V_j}, M, T_{V_i}, K_{V_i} \oplus H(M||T_{V_i})\}$. However, it is infeasible to derive the symmetric key K_{V_i} because the attacker does not know the secret key x of the RSU, where $K_{V_i} = H(ID_{V_i}||x)$. To compute K_{V_i}, the attacker has to know the secret key x. Hence, the outsider attack is impossible for the proposed scheme.

5 Conclusions

In this paper, we propose an efficient authenticated communication scheme for the traffic control on the VANET. In the proposed scheme, a vehicle communicates with the other vehicles through RSU. Based upon this idea, a vehicle only needs to share one session key with RSU to communicate with the other vehicles in the proposed schemes. In addition, the communication model of the proposed schemes is simpler than that of Li et al.'s schemes. Therefore, the proposed schemes are more efficient and practical than the previously proposed schemes for the VANET. In the future, we will investigate a new communication scheme without using the elliptic curve cryptosystem so the vehicle communications on the VANET can become more efficient in practice.

References

[1] M. S. Bouassida, I. Chrisment, and O. Festor, "Group key management in MANETs," *International Journal of Network Security*, vol. 6, no. 1, pp. 67–79, 2008.

[2] D. Chaum, "Blind signature systems," in *Proceedings of Advances in Crypto'83*, pp. 153, 1983.
[3] A. K. Das, "An identity-based random key predistribution scheme for direct key establishment to prevent attacks in wireless sensor networks," *International Journal of Network Security*, vol. 6, no. 2, pp. 129–139, 2008.
[4] T. ElGmal, "A public-key cryptosystem and a signature scheme based on discrete logarithms," *IEEE Transactions on Information Theory*, vol. 31, no. 4, pp.469–472, 1985.
[5] S. Eichler, F. Dotzer, C. Schwingenschologl, F. J. F. Vehicleo, and J. Eberspacher, "Secure routing in a vehicular ad hoc network," in *Proceedings of IEEE 60th Vehicular Technology Conference*, pp. 3339–3343, 2004.
[6] M. S. Farash and M. A. Attari, "A pairing-free ID-based key agreement protocol with different PKGs," *International Journal of Network Security*, vol. 16, pp. 144–149, 2014.
[7] D. Johnson, A. Menezes, and S. Vanstone, *The Elliptic Curve Digital Signature Algorithm (ECDSA)*, Techical Report CORR 99-34, 1999.
[8] J. Kar, "ID-based deniable authentication protocol based on Diffie-Hellman problem on elliptic curve," *International Journal of Network Security*, vol. 15, pp. 347–354, 2013.
[9] N. Koblitz, A. J. Menezes, and S. A. Vanstone, "The state of elliptic curve cryptography," *Design, Codes and Cryptography*, vol. 19, no. 2-3, pp. 173–93, 2000.
[10] J. S. Lee and C. C. Chang, "Secure communications for cluster-based ad hoc networks using node identities," *Journal of Network and Computer Applications*, vol. 30, no. 4, pp. 1377–1396, 2006.
[11] C. T. Li, M. S. Hwang, and 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, 2008.
[12] T. Leinmuller, C. Maihofer, E. Schoch, and F. Karql, "Improved security in geographic ad hoc routing through autonomous position verification," in *Proceedings of the 3rd International Workshop on Vehicular Ad Hoc Networks*, pp.57–66, 2006.
[13] T. Leinmuller, E. Schoch, and F. Karql, "Position verification approaches for vehicular ad hoc networks," *IEEE Wireless Communications*, vol. 13, no. 5, pp. 16–21, 2006.
[14] J. V. D. Merwe, D. Dawoud, and S. Mcdonald, "A survey on peer-to-peer key management for mobile ad hoc networks," *ACM Computing Surveys*, vol. 39, no. 1, pp. 1–45, 2007.
[15] U. M. Maurer and Y. Yacobi, "A non-interactive public-key distribution system," *Design, Codes and Cryptography*, vol. 9, no. 3, pp. 305–316, 1996.
[16] K. Plossl, T. Nowey, and C. Mletzko, "Towards a security architecture for vehicular ad hoc networks," in *Proceedings of the First International Conference on Availability, Reliability and Security*, pp. 374–381, 2006.
[17] W. Ren, "Pulsing RoQ DDoS attacking and defense scheme in mobile ad hoc networks," *International Journal of Network Security*, vol. 4, no. 2, pp. 227–234, 2007.
[18] R. L. Rivest, A. Shamir, and L. Adleman, "A method for obtaining digital signatures and public key cryptosystems," *Communications of the ACM*, vol. 21, no. 2, pp. 120–126, 1978.
[19] B. Sieka and A. D. Kshemkalyani, "Establishing authenticated channels and secure identifiers in ad-hoc networks," *International Journal of Network Security*, vol. 5, no. 1, pp. 51–61, 2007.
[20] Y. M. Tseng and J. K. Jan, "ID-based cryptographic scheme using a non-interactive public-key distribution system," in *Proceedings of the 14th Annual Computer Security Applications Conference (IEEE ACSAC'98)*, pp. 237–243, 1998.
[21] Y. Zhang, W. Liu, W. Lou, and Y. Fang, "Securing mobile ad hoc networks with certificateless public keys," *IEEE Transactions on Dependable and Secure Computing*, vol. 3, no. 4, pp.386–399, 2006.

Chin-Chen Chang received his Ph.D. degree in computer engineering from National Chiao Tung University. His first degree is Bachelor of Science in Applied Mathematics and master degree is Master of Science in computer and decision sciences. Both were awarded in National Tsing Hua University. Dr. Chang served in National Chung Cheng University from 1989 to 2005. His current title is Chair Professor in Department of Information Engineering and Computer Science, Feng Chia University, from Feb. 2005. He is currently a Fellow of IEEE and a Fellow of IEE, UK. His current research interests include database design, computer cryptography, image compression and data structures.

Jen-Ho Yang received the BS degree in Computer Science and Information Engineering from I-Shou University, Kaoshiung, Taiwan in 2002. He received his Ph.D. degree in Computer Science and Information Engineering from National Chung Cheng University, Chiayi County, Taiwan in 2009. Since 2009, he has been an assistant professor of Department of Multimedia and Mobile Commerce in Kainan University, Taoyuan County, Taiwan. His current research interests include electronic commerce, information security, cryptography, authentication mechanisms, digital right management, and fast modular multiplication algorithm.

Yu-Ching Wu received the BS degree in Computer Science and Information Engineering from National Chung Cheng University, Chiayi County, Taiwan in 2009. His current research interests include information security, cryptography, and vehicular ad hoc networks.

On the Privacy of "User Efficient Recoverable Off-Line E-Cash Scheme with Fast Anonymity Revoking"

Yalin Chen[1] and Jue-Sam Chou[2]
(Corresponding author: Jue-Sam Chou)
Institute of information systems and applications, National Tsing Hua University, Taiwan[1]
Department of Information Management, Nanhua University, Taiwan[2]
No. 55, Sec. 1, Nanhua Rd., Dalin Township, Chiayi County 62249, Taiwan (R.O.C.)
(Email: jschou@mail.nhu.edu.tw)
(Received Dec. 3, 2014; revised and accepted Apr. 16 & May 26, 2015)

Abstract

Recently, Fan et al. proposed a novel e-cash scheme which allows a user to recover the e-cash he lost. They claimed their e-cash possesses properties of anonymity, unlinkability (i.e. untraceability), bank-off-line payment, double-spending detection, and anonymity revocation. The e-cash untraceability is greatly related to users' privacy and indicates that no one including the issuer bank can link e-cash to any user when the e-cash is legally spent. Although, the authors have formally proved the unlinkability of their scheme, we still found a loophole to compromise user's privacy. That is, an issuer bank or an attacker who intrudes the issuer bank's system can link e-cash to a user by collecting e-cash withdrawal and deposit transaction messages. This may make the user's shopping behaviors or location information exposed.

Keywords: Anonymity revocation, digital signatures, electronic commerce and payment, off-Line E-Cash, recoverable, RSA

1 Introduction

With the advances of technology, people are paying through diverse payment tools or systems [2, 8, 16, 18], for example credit cards, debit cards, PayPal, account transfer, Short-Message-Service (SMS) payment, mobile phone payments, electronic transportation toll, and etc. Most of the payment tools or systems are named payments which make the payers' identities exposed to the brokers or intermediaries. In the case of the globally widespread credit card payments, card-issuing banks are aware of the contents of the cardholder's all spending, such as the cardholder went somewhere to buy something at some time, visited some restaurant to have dinner, or travelled to some gas station, and so forth. To prevent personal privacy exposure to the payment intermediaries, electronic cash (e-cash) which holds the anonymity property like dollar bill can make the payer not to be aware of and not to be tracked. There have been many cryptographic scientists working within the field of e-cash system design [1, 3, 4, 5, 6, 7, 9, 10, 11, 17] since Chaum first proposed the concept of e-cash in 1982. From the viewpoint of control, e-cash systems fall into two categories: bank-controlled e-cash systems, ex. Mondex, and P2P (peer-to-peer)-distributed e-cash systems, ex. Bitcoin.

Mondex [15] developed by National Westminster Bank in the U. K. and had big success in 1990s. It has the advantage of absolute anonymity but opens a perfect way for criminals to illegally transfer funds with untraceability. While Bitcoin [13, 19] kills the role of the central bank or authority, reduces the expensive bank-processing cost, and prevails over the cyberspace and the real world. All activities including coin mintage, coin validness check, double-spending check are done through the cooperation of the peer nodes on the Bitcoin P2P network. By just generating a public/private key pair, a user can join the Bitcoin network, and he/she uses this public key as a his/her pseudonym to mine, exchange, buy, and pay the Bitcoin without revealing his/her real identity and location. However, some privacy issues exist since all Bitcoin transactions are public. One may trace sensitive transactions or de-anonymize social network data using network topology, thus violating users' privacy [12, 14].

For a sound cash system, some essential properties should be focused.

Verifiability. The validness of e-cash can be publicly examined.

Unforgeability. E-cash should be only issued through defined procedures. No one including banks can forge e-cash by other ways.

Anonymity. It indicates that a user need not present his/her real identity when paying.

Untraceability or Unlinkability. It means that no one including the banks can know the owner of the e-cash when it is legally used. Specially, although the bank provides e-cash withdrawal service to her account holder, it cannot link any e-cash to her account holder.

Double-spending detection. An e-cash system should prevent e-cash to be spent twice. If it happens, the system should efficiently catch the cheater.

Anonymity revocation. When e-cash is used for illegal purposes such as money laundering and tax evading, the system should disclose the owner identity of the e-cash.

Recently, Fan et al. [3] proposed a bank-off-line e-cash scheme with fast anonymity revoking. Bank-off-line e-cash indicates that e-cash in a payment transaction need not an online bank to examine its validness. Fan et al. claimed that each user possessed anonymity and unlinkability, and the user is allowed to recover his e-cash when lost. Besides, the bank can detect the double spending and efficiently derive the identity of the user, without any help of the Trust Third Party (TTP). Moreover, TTP can revoke the anonymity of the e-cash owner when illegal transaction occurs. Additionally, Fan et al. scheme allows the police to trace a specific user. However, after examining their scheme, we found that it does not have anonymity and unlinkability.

2 Review of Fan et al.'s Scheme

Fan et al.'s e-cash scheme [3] applies the concepts of Chaum's blind signature and the chameleon hashing function. It consists of two main protocols, the withdrawal protocol and the payment protocol, and four entities user, bank, shop and the judge. The bank publishes $\{n_b, e_b\}$ as RSA public key, H as one-way hash function, and $\{p, q, g\}$ as the parameters of chameleon hashing function h_{Hk}. The judge generates public and private key pair $\{pk_j, sk_j\}$, and embeds $\{pk_j, sk_j, H, h_{Hk}, n_b, e_b\}$ into a tamper-resistant device. In the below, we first describe Fan et al.'s withdrawal and payment protocols and show their weaknesses then.

2.1 E-cash Withdrawal

In an e-cash withdrawal process, Fan et al.'s scheme assumes that the bank authenticates her account holder through a secure channel first. The bank and the user then perform the following e-cash withdrawal protocol.

1) User → Bank: $E_{pk_j}(k, m, r)$.
The user randomly chooses three random strings $\{k, m, r\}$ and he then sends $E_{pk_j}(k, m, r)$ to the bank, where $E(\cdot)$ is a public key encryption algorithm and k is a secret session key to be shared with the judge's device embedded in the bank system.

2) Bank → Judge's device: $\{E_{pk_j}(k, m, r), \mu\}$.
After the bank authenticates the user, she knows the user's identity ID_u. She then sends $E_{pk_j}(k, m, r)$ and $\mu = ID_u$ to the judge's device.

3) Judge's device → Bank: $\{\beta, E_k(x, x', c, k, \delta)\}$.
On receiving $\{E_{pk_j}(k, m, r), \mu\}$, the judge's device uses the stored private key sk_j to decrypt $E_{pk_j}(k, m, r)$ and obtain $\{k, m, r\}$. Then, it randomly chooses three strings $\{r_1, r_2, c\}$, and computes

$$\begin{aligned} x &= (\mu||r_1), \\ x' &= x^{-1} (\mathrm{mod} q), \\ \delta &= E_{pk_j}(\mu, r_2), \\ y &= g^x (\mathrm{mod} p), \\ \beta &= (c^{-1})^{e_b} h_{Hk}(m, r) H(\delta||y) (\mathrm{mod} n_b), \end{aligned}$$

where $h_{Hk}(m, r) = g^m y^r (\mathrm{mod} q)$, and outputs $\{\beta, E_k(x, x', c, k, \delta)\}$ to the bank system. Note that $h_{Hk}(\cdot)$ is a chameleon hashing function with the secret key Hk; given $h_{Hk}(m, r)$ one can easily find a preimage (m', r') such that $h_{Hk}(m', r') = h_{Hk}(m, r)$ if he knows the secret key Hk.

4) Bank → User: $\{t, E_k(x, x', c, k, \delta)\}$
On receiving the device response, the bank system returns $\{t = \beta^{d_b}, E_k(x, x', c, k, \delta)\}$ to the user (where d_b is the bank's RSA signing key), and stores $\{ID_u, E_{pk_j}(k, m, r), E_k(x, x', c, k, \delta)\}$ in her database.

5) User unblinding e-cash On receiving $\{t, E_k(x, x', c, k, \delta)\}$, the user decrypts $E_k(x, x', c, k, \delta)$ and parses the 4th parameter in the decryption result as k'. Then he checks whether $k' = k$. If it holds, he computes $\sum = ct (\mathrm{mod} n_b)$. At last, the user obtains an e-cash as $\{\sum, y, m, r, \delta\}$.

2.2 E-cash Paying and Deposit

A user in Fan et al.'s e-cash system allows to pay his cash to an Internet shop in the bank-offline manner as follows.

1) Shop → User: $\{m'\}$.
On receiving the user's payment request, the shop sends a challenge $m' = (ID_s r_s)$ to the user, where ID_s is the shop's identity and r_s is a random string.

2) User → Shop: $\{\sum, y, r', \delta\}$
On receiving the shop's challenge m', the user computes $r' = x'(m + xr - m') (\mathrm{mod} q)$ and answers the shop $\{\sum, y, r', \delta\}$.

3) Shop → Bank: $\{\sum, y, m', r', \delta\}$ On receiving the user's response, the shop verifies if the following equation holds.

$$\sum^{e_b} = h_{Hk}(m', r') H(\delta||y) (\mathrm{mod} n_b).$$

If it holds, the shop accepts the e-cash and stores the e-cash transcript $\{\sum, y, m', r', \delta\}$. In the clear time, the shop sends the bank the e-cash transcript.

4) Bank: acceptance or rejection. On receiving the e-cash transcript for deposit from the shop, the bank first verifies the e-cash by checking if the equation in 3) holds and if data $\{\sum, y, \delta\}$ has not existed in bank's database. If both are true, the bank stores the e-cash transcript $\{\sum, y, m', r', \delta\}$ in the database and credits the shop's account. Otherwise, the bank rejects the shop's deposit request.

3 A Loophole of Users' Privacy

An attacker can collect the transmitted messages of withdrawal, payment and deposit transactions in Fan et al.'s e-cash system, and obtain information as follows:

1) From a withdrawal transaction, the attacker can know the values, μ, β, and t. Notice that the user in the end of the transaction privately produces the e-cash $\{\sum, y, m, r, \delta\}$ which is not known to any other ones including the attacker.

2) From an off-line payment transaction, the attacker can know the e-cash, $\{\sum^*, y^*, m^*, r^*, \delta^*\}$ from the communication.

He then launches an offline attack by the following steps.

1) Computes $c^* = \sum^* t^{-1} (\mathrm{mod} n_b)$.

2) Computes to see if $\beta \stackrel{?}{=} h_{Hk}(m^*, r^*) H(\delta^* || y^*)$.

If the equation in 2) holds, the attacker links the e-cash $\{\sum^*, y^*, m^*, r^*, \delta^*\}$ to the user whose identity is $\mu (= IDu)$. Thus, the features of anonymity and unlinkability are broken.

4 Conclusion

In this paper, we showed that Fan et al.'s recoverable off-line e-cash's scheme is flawed. It suffers from linkability and identity leakage. This may result in e-cash user's shopping behaviors and movement information exposed to banks or attackers.

References

[1] M. Z. Ashrafi, S. K. Ng, "Privacy-preserving e-payments using one-time payment details", *Computer Standards & Interfaces*, vol. 31, no. 2, pp. 321–328, 2009.

[2] L. Aszalós, A. Huszti, "Payment approval for PayWord", in *Information Security Applications*, pp. 161–176, 2012.

[3] Y. Baseri, B. Takhtaei, and J. Mohajeri, "Secure untraceable off-line electronic cash system", *Scientia Iranica*, vol. 20, no. 3, pp. 637–646, 2013.

[4] J. Camenisch, U. Maurer, M. Stadler, "Digital payment systems with passive anonymity-revoking trustees", *Journal of Computer Security*, vol. 5, no. 1, pp. 69–89, 1997.

[5] D. Chaum, "Blind signatures for untraceable payments", in *Advances in Cryptology (Crypto'82)*, LNCS 82, pp. 199–203, 1983.

[6] D. Chaum, A. Fiat, M. Naor, "Untraceable electronic cash", in *Advances in Cryptology (Crypto'88)*, pp. 319–327, 1988.

[7] Y. Chen, J. S. Chou, H. M. Sun, M. H. Cho, "A novel electronic cash system with trustee-based anonymity revocation from pairing", *Electronic Commerce Research and Applications*, vol. 10, no. 6, pp. 673–682, 2011.

[8] K. K. R. Choo, "New payment methods", *Computers & Security*, vol. 36, pp. 12–26, 2013.

[9] Z. Eslami, M. Talebi, "A new untraceable off-line electronic cash system", *Electronic Commerce Research and Applications*, vol. 10, no. 1, pp. 59–66, 2011.

[10] C. I. Fan, V. S. M. Huang, Y. C. Yu, "User efficient recoverable off-line e-cash scheme with fast anonymity revoking", *Mathematical and Computer Modelling*, vol. 58, no. 1-2, pp. 227–237, 2012.

[11] W. S. Juang, "D-cash: A flexible pre-paid e-cash scheme for date-attachment", *Electronic Commerce Research and Applications*, vol. 6, no. 1, pp. 74–80, 2007.

[12] I. Miers, C. Garman, M. Green, A. D. Rubin, "Zerocoin: Anonymous distributed e-cash from bitcoin", in *IEEE Symposium on Security and Privacy*, pp. 397–411, 2013.

[13] M. E. Peck, "The cryptoanarchists' answer to cash", *IEEE Spectrum*, vol. 49, no. 6, pp. 50–56, 2012.

[14] F. Reid, M. Harrigan, "An analysis of anonymity in the bitcoin system", in *Security and Privacy in Social Networks*, pp. 197–223, 2013.

[15] F. Stalder, "Failures and successes: Notes on the development of electronic cash", *The Information Society: An International Journal*, vol. 18, no. 3, pp. 209–219, 2002.

[16] G. W. H. Tan, K. B. Ooi, S. C. Chong, T. S. Hew, "NFC Mobile Credit Card: The Next Frontier of Mobile Payment?", *Telematics and Informatics*, vol. 31, pp. 292–307, 2014.

[17] H. Wang, J. Cao, Y. Zhang, "A flexible payment scheme and its role-based access control", *IEEE Transactions on Knowledge and Data Engineering*, vol. 17, no. 3, pp. 425–436, 2005.

[18] Q. Wang, J. Zhu, "Study on the Electronic Payment Technology in E-Commerce", in *Proceedings of the 2nd International Conference on Green Communications and Networks (GCN'12)*, pp. 95–100, 2013.

[19] G. Zorpette, "The beginning of the end of cash", *IEEE Spectrum*, vol. 49, no. 6, pp. 27–29, 2012.

Yalin Chen received his Ph.D. degree in the Institute of Information Systems and Applications from National Tsing Hua University (NTHU) in Hsinchu, Taiwan, ROC. She now hosts the C & C Information Security Laboratory with Dr. Jue-Sam Chou in Chiayi, Taiwan. She now is also a contributing editor of Journal of Computer Science. Her primary research interests are Information Security, Cryptographic Protocols, Data security and Privacy, Authentication, Key Agreement, Electronic Commerce Security, E-commerce Protocols, Ad-Hoc Network Security, Sensor Network Security, RFID Authentication Protocol, Electronic Cash, Electronic Voting.

Jue-Sam Chou received his Ph.D. degree in the department of computer science and information engineering from National Chiao Tung Univ. (NCTU) in Hsinchu, Taiwan,ROC. He is an associate professor and teaches at the dept. of Infomation Management of Nanhua Univ. in Chiayi, Taiwan. He is now editors of two journals: Journal of computer science and International Journal of Cognitive Research in Science and Engineering and Education (IJCRSEE). His primary research interests are Information Security, Cryptographic Protocols, Data security and Privacy, Authentication, Key Agreement, Electronic Commerce Security, E-commerce Protocols, Ad-Hoc Network Security, Sensor Network Security, RFID Authentication Protocol, Electronic Cash, Electronic Voting.

A Safety Review on Fuzzy-based Relay Selection in Wireless Sensor Networks

Tung-Huang Feng[1], Neng-Yih Shih[1], and Min-Shiang Hwang[1,2]
(Corresponding author: Min-Shiang Hwang)
Department of Computer Science and Information Engineering, Asia University[1]
No. 500, Lioufeng Raod, Wufeng Shiang, Taichung, Taiwan (R.O.C.)
Department of Medical Research, China Medical University Hospital, China Medical University[2]
(Email: mshwang@asia.edu.tw)
(Received Mar. 13, 2015; revised and accepted May 2 & June 4, 2015)

Abstract

A Wi-Fi has been able to accomplish great transmission due to its co-transfer mode. Each wireless network node has to rely on the way to achieve its mission of information transmission. In unattended sensing network environment, the relay process of how to select an appropriate relay nodes can adjust power consumption and communication quality, so it has attracted a lot of research attention and considerable results. Yang and Brante are few of researchers in recent years. They applied the fuzzy inference system to the relay selection algorithm, which was confirmed to have a good performance. In this study, a safety assessment method for fuzzy relay selection algorithms has been introduced. and introduce the safety performance of Yang's and Brante's method. Then this article can be expected to extend the depth and breadth by the future research.

Keywords: Fuzzy relationship rule, relay selection algorithm, wireless sensor network

1 Introduction

This study focuses on the issue of saving sensor networks. The need for energy-saving of sensor networks is required, because it was difficult or impossible to charge energy in some application areas at unattended sensing nodes. In this case, the whole sensing network life cycle was dependent on battery power [24, 31]. In the energy-saving study area, it is usually divided into three areas: sensing, aggregated data and communication [11, 12]. The relay selection algorithms belong to the field of communications. The research in this area pointed out that the biggest action of power consumption in the wireless sensor nodes are transmitted and received [3], and most effective energy-saving strategy is to use sleep mode [7]. This strategy allows a node only to transmit and receive when it?s necessary. However, how to select the most appropriate partners [1] from candidate relay gateway nodes in staggered complex wireless sensor networks is to avoid transmitting and receiving the same information by all possible nodes. Repeating each other's electricity consumption will result in lower overall network life cycle. That was still a very important issue [6, 31]. Specifically, the trade-off problem [2] was how to balance the communication quality and remainder power between nodes, and still able to maintain optimal network life cycle.

Cover and ElGamal proposed the concept of three relay architecture [5] first in 1979. The existing study of relay selection techniques can be classified as a measure, effectiveness threshold and adjustment patterns opportunity. For communication nodes to exchange information point of view, it can be divided into two types of forms and competitions [23]. Recently, for the fuzzy relay selection algorithm, it is based on fuzzy sets to use fuzzy rule base and defuzzification architecture out of the relay selection algorithms. Yang made the first algorithm based fuzzy relay selection, and proved the effectiveness of the use of fuzzy theory that can make performers get traditional relay selection algorithm [28]. Brante structure is a more complete fuzzy relay selection algorithm. It focuses on the balance of nodes between the quality of communication and remainder power, and optimizing the efficiency of the algorithm [4]. This paper suggests that poor design of fuzzy relay selection algorithms is most likely to suffer an attack [22]. If the algorithm without adjustment, then it would choose a high power, high communication state parameters. It will come to the opposite effect, and lose the purposes to reach energy-saving and adjust communication in a power attack mode. Therefore, that needs to be properly analyzed to protect its safety.

This paper are organized as follows: Section 2 describes the relay selection algorithms, including the fuzzy relay selection algorithms proposed by Yang and Brante. Section 3 describes a safety test system. Section 4 is to analyze the results of Yang and Brante algorithm in a safety test system. Section 5 is about future research. The last

section is to make a conclusion.

2 Fuzzy Relay Selection Algorithm

Yang [28] and Brante [4] proposed a fuzzy relay selection algorithm. The algorithm has shown better trade-off than traditional effect. This section describes a fuzzy system architecture of wireless sensor networks, and the research methods proposed by Yang and Brante who used the fuzzy rule base system by this architecture.

2.1 Fuzzy Based Relay Selection Algorithm

The fuzzy inference system has first started since 1965. When Zadeh published the fuzzy logic and created the fuzzy theory [29, 30]. It has been developed over the years. Those theoretical methods included: the membership function, fuzzy, fuzzy inference rule and defuzzification [21]. The well known MATLAB platform has built based on the Foundations of Fuzzy Logic to specificly practice Mamdani [17, 18] and Sugeno's theory [17, 26]. Moreover, MATLAB provides an easy-to-operate graphical interface [10] with Fuzzy Logic Toolbox. Such a complete modeling environment has become a favorite tool of researchers [25].

A typical fuzzy inference system architecture built on MATLAB as in Figure 1.

Figure 1: A block diagram of a typical fuzzy inference system

A simple instance exercises of typical fuzzy inference system on MATLAB is shown in Figure 2. It accounts for the situation of fuzzy relay selection algorithm on wireless sensor networks. The construction of its contents is shown in the following subsections.

2.1.1 Define the Variables

This example is a scenarios for battery-powered wireless sensor networks. The main application of the fuzzy controller is to regulate power consumption and communication status. Therefore, Two input variables are defined as the *Re* (remaining energy) and *CSI* (Chanel State Information). One output variable is defined as Select (probability of a candidate node for forwards select). Its representations are as follows: (1) variable domain range, (2) variable parameter, (3) parameter semantic and (4) member function.

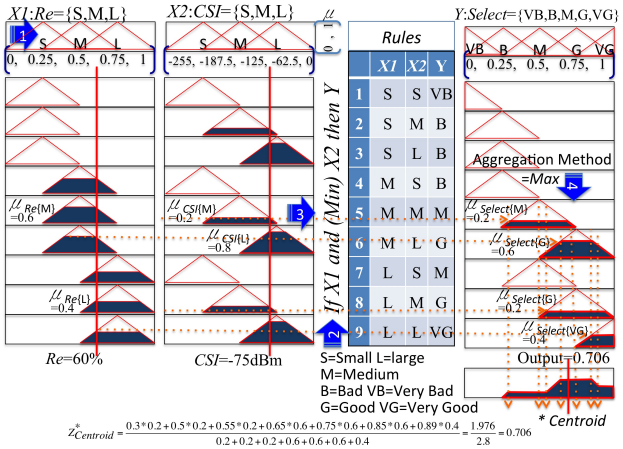

Figure 2: Typical instance of a fuzzy inference system on MATLAB

Variable Domain Range:

$$\begin{aligned} Re = &\quad [0 \sim 100\%] \\ CSI = &\quad [-255dBm \sim 0dBm] \\ Select = &\quad [0 \sim 100\%] \end{aligned}$$

Variable Parameter:

$$\begin{aligned} Re = &\quad \{\mathbf{S} : Small, \mathbf{M} : Medium, \mathbf{L} : Large\} \\ CSI = &\quad \{\mathbf{S} : Small, \mathbf{M} : Medium, \mathbf{L} : Large\} \\ Select = &\quad \{\mathbf{VB} : VeryBad, \mathbf{B} : Bad, \\ &\quad \mathbf{M} : Medium, \\ &\quad \mathbf{G} : Good, \mathbf{VG} : VeryGood\} \end{aligned}$$

Parameter Semantic:

$$\begin{aligned} Re = &\quad \{Lesspower, Mediumpower, Highpower\} \\ CSI = &\quad \{Badsignal, Mediumsignal, Goodsignal\} \\ Select = &\quad \{Verybad, Bad, InterMedium, Good, \\ &\quad Verygood\} \end{aligned}$$

Member Function:

$$\begin{aligned} Re\{\mathbf{S}\} = &\quad trimf(X1 : 0, 0.25, 0.5) \\ Re\{\mathbf{M}\} = &\quad trimf(X1 : 0.25, 0.5, 0.75) \\ Re\{\mathbf{L}\} = &\quad trimf(X1 : 0.5, 0.75, 1) \\ CSI\{\mathbf{S}\} = &\quad trimf(X2 : -255, -187.5, -125) \\ CSI\{\mathbf{M}\} = &\quad trimf(X2 : -187.5, -125, -62.5) \\ CSI\{\mathbf{L}\} = &\quad trimf(X2 : -125, -62.5, 0) \\ Select\{\mathbf{VB}\} = &\quad trimf(Y : 0, 0, 0.25) \\ Select\{\mathbf{B}\} = &\quad trimf(Y : 0, 0.25, 0.5) \\ Select\{\mathbf{M}\} = &\quad trimf(Y : 0.25, 0.5, 0.75) \\ Select\{\mathbf{G}\} = &\quad trimf(Y : 0.5, 0.75, 1) \\ Select\{\mathbf{VG}\} = &\quad trimf(Y : 0.75, 1, 1) \end{aligned} \tag{1}$$

This example used Triangle Membership function (trimf). The input values will belong to a different set, and convert to the μ (membership grade) between 0∼1.

2.1.2 Design a Fuzzy Rule Base

A fuzzy rule base is based on experience and expert knowledge. It's translated to semantic with the control rules as "If X1 and X2 Then Y". The example was divided into three fuzzy parameters. Therefore, it can be deduced to a relation matrix of $3 * 3$ input and output in Table 1. Thus, the rule bases are got as nine fuzzy rules:

$$\begin{aligned}
&Rule1: If\ X1: Re\{S\}\ and\ X2: CSI\{S\}\\
&\quad Then\ Y: Select\{VB\}\\
&Rule2: If\ X1: Re\{S\}\ and\ X2: CSI\{M\}\\
&\quad Then\ Y: Select\{B\}\\
&Rule3: If\ X1: Re\{S\}\ and\ X2: CSI\{L\}\\
&\quad Then\ Y: Select\{B\}\\
&Rule4: If\ X1: Re\{M\}\ and\ X2: CSI\{S\}\\
&\quad Then\ Y: Select\{B\}\\
&Rule5: If\ X1: Re\{M\}\ and\ X2: CSI\{M\}\\
&\quad Then\ Y: Select\{M\}\\
&Rule6: If\ X1: Re\{M\}\ and\ X2: CSI\{L\}\\
&\quad Then\ Y: Select\{G\}\\
&Rule7: If\ X1: Re\{L\}\ and\ X2: CSI\{S\}\\
&\quad Then\ Y: Select\{M\}\\
&Rule8: If\ X1: Re\{L\}\ and\ X2: CSI\{M\}\\
&\quad Then\ Y: Select\{G\}\\
&Rule9: If\ X1: Re\{L\}\ and\ X2: CSI\{L\}\\
&\quad Then\ Y: Select\{VG\}
\end{aligned} \tag{2}$$

Table 1: The relation matrix of fuzzy rules base

Re**CSI	**S**	**M**	**L**
S	VB	B	B
M	B	M	G
L	M	G	VG

2.1.3 Fuzzification

Assume the input value is set to $Re = 60\%$, CSI = -75dBm in Figure 2. According to the membership function represented by Equations (1) and (2), the membership grade is obtained: $\mu_{Re\{M\}} = 0.6$, $\mu_{Re\{L\}} = 0.4$, $\mu_{CSI\{M\}} = 0.2$, $\mu_{CSI\{L\}} = 0.8$. Next, the specified rule is based on this instance process Min fuzzy arithmetic with the And of sets operator to output membership function value as shown in Table 2:

2.1.4 Defuzzification

It's not the only value was obtained by the output membership function. In this case, calculated result was using the Max aggregation method and the Centroid method to defuzzification such as follows:

$$\begin{aligned}
Z^*_{Cente} &= \frac{Z^1 + Z^2 + Z^3 + Z^4 + Z^5 + Z^6 + Z^7}{0.2 + 0.2 + 0.2 + 0.6 + 0.6 + 0.6 + 0.4}\\
&= \frac{1.976}{2.8}\\
&= 0.706
\end{aligned} \tag{3}$$

Notes:

$$\begin{aligned}
Z^1 &= 0.3 \times 0.2,\\
Z^2 &= 0.5 \times 0.2,\\
Z^3 &= 0.55 \times 0.2,\\
Z^4 &= 0.65 \times 0.6,\\
Z^5 &= 0.75 \times 0.6,\\
Z^6 &= 0.85 \times 0.6,\\
Z^7 &= 0.85 \times 0.6,\\
Z^8 &= 0.89 \times 0.4.
\end{aligned}$$

The output value of 0.706 was got by the typical fuzzy inference system. If the value is greater than the defuzzification output of other nodes, eventually, this node is selected as the relay node from fuzzy relay selection algorithms in wireless sensor networks.

2.2 Yang's Fuzzy Inference Method

Although fuzzy theory have developed for many years. But it can be applied to regulate power consumption and communication status of a relay selection method in wireless sensor networks only by Yang until 2009. Although the relay selection algorithm by Yang's have assigned the membership function within fuzzy system functions. But it's only using the traditional method of weighted average instead of the fuzzy inference rule and defuzzification process. Yang's fuzzy theory operation is shown in the following subsections.

2.2.1 Define the Variables

Variable Domain Range:

$$\begin{aligned}
Re &= [0 \sim 100\%]\\
CSI &= [-255dBm \sim 0dBm].
\end{aligned}$$

Variable Parameter:

$$\begin{aligned}
Re &= \{\mathbf{All}\},\\
CSI &= \{\mathbf{All}\}.
\end{aligned}$$

Parameter Semantic:

$$\begin{aligned}
Re &= \{\mathbf{Power}\},\\
CSI &= \{\mathbf{Signal}\}.
\end{aligned}$$

Member Function:

$$\begin{aligned}
Re\{\mathbf{All}\} &= trimf(X1: 0, 0, 1)\\
CSI\{\mathbf{All}\} &= trimf(X2: -255, -255, 0).
\end{aligned}$$

2.2.2 Design a Fuzzy Rule Base

From the above, Yang uses a single variable parameter. The fuzzy system will create the equivalent of fuzzy input matrix R:

$$R = [\mu_{Re\{\mathbf{All}\}}, \mu_{CSI\{\mathbf{All}\}}].$$

Table 2: Operation results by Min fuzzy method

Re* \ *CSI	$\mu_{CSI\{S\}} = 0$	$\mu_{CSI\{M\}} = 0.2$	$\mu_{CSI\{L\}} = 0.8$
$\mu_{Re\{S\}} = 0$	Min(0,0)=0	Min(0,0.6)=0	Min(0,0.8)=0
$\mu_{Re\{M\}} = 0.6$	Min(0.6,0)=0	Min(0.6,0.2)= $\mu_{Select\{M\}}(0.2)$	Min(0.6,0.8)= $\mu_{Select\{G\}}(0.6)$
$\mu_{Re\{L\}} = 0.4$	Min(0.4,0)=0	Min(0.4,0.2)= $\mu_{Select\{G\}}(0.2)$	Min(0.4,0.8)= $\mu_{Select\{VG\}}(0.4)$

2.2.3 Fuzzification and Defuzzification

Yang uses the output weights W as a defuzzification operation mechanism:

$$W = (W_{Re} = 0.5, W_{CSI} = 0.5).$$

Finally, the following equation is used to assess the output results to determine a relay node with a maximum value:

$$\begin{aligned} Z &= R \bullet W \\ &= (\mu_{Re\{\mathbf{All}\}} \times W_{Re}) + (\mu_{CSI\{\mathbf{All}\}} \times W_{CSI}). \end{aligned}$$

The simulation of Yang's relay selection algorithm have proved that the network life cycle is improved than a relay selection algorithms which only assess a single network quality with *CSI*. Although Yang method is still using some of the traditional method, the survey of literature [1] for relay selection algorithms has not still categorized the fuzzy system relay selection algorithms as an exclusive categories; Brante said: Yang's research is the first fuzzy system to use relay selection algorithms [4].

2.3 Brante's Fuzzy Inference Method

Brante proposed a distributed relay selection algorithm on wireless sensor networks in 2013. Relative to Yang relay selection algorithm in 2009, Brante's research has been able to use the complete fuzzy systems to control the selection of cooperative nodes. The block diagram of Brante's fuzzy system for relay selection algorithms is shown in Figure 3. Brante's fuzzy theory operated is shown in the following subsections.

2.3.1 Define the Variables

Variable Domain Range:

$$\begin{aligned} Re &= [0 \sim 100\%] \\ CSI &= [0 \sim \infty]. \end{aligned}$$

Variable Parameter:

$$\begin{aligned} Re &= \{L : Low; M : Medium; F : Full\} \\ CSI &= \{W : Weak; A : Average; S : Strong\} \\ Select &= \{VB : VeryBad; B : Bad; \\ &\quad M : Medium; G : Good; \\ &\quad VG : VeryGood\} \end{aligned}$$

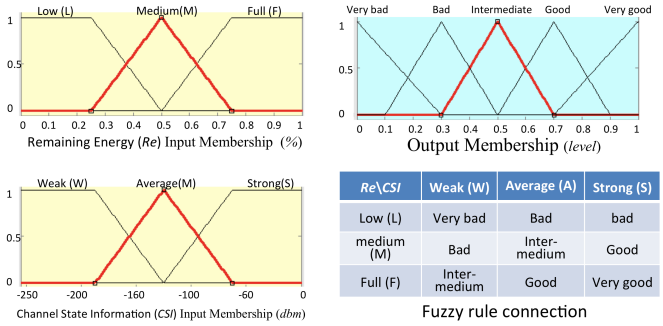

Figure 3: The block diagram of Brante's fuzzy system for relay selection

Parameter Semantic:

$$\begin{aligned} Re &= \{Less\ Power, Medium\ Power, High\ Power\} \\ CSI &= \{Bad\ Signal, Medium\ Signal, Good\ Signal\} \\ Select &= \{VeryBad, \\ &\quad Bad, \\ &\quad Medium, \\ &\quad Good, \\ &\quad VeryGood\}. \end{aligned}$$

Member Function: Use Trapezoid Membership function (trapmf).

$$\begin{aligned} Re\{\mathbf{L}\} &= trapmf(X1 : 0, 0, 0.25, 0.5) \\ Re\{\mathbf{M}\} &= trapmf(X1 : 0.25, 0.5, 0.5, 0.75) \\ Re\{\mathbf{F}\} &= trapmf(X1 : 0.5, 0, 75, 1, 1) \\ CSI\{\mathbf{W}\} &= trapmf(X2 : -255, -255, -187.5, \\ &\qquad -125) \\ CSI\{\mathbf{A}\} &= trapmf(X2 : -187.5, -125, -125, \\ &\qquad -62.5) \\ CSI\{\mathbf{S}\} &= trapmf(X2 : -125, -62.5, 0, 0) \\ Select\{\mathbf{VB}\} &= trapmf(Y : 0, 0, 0, 0.3) \\ Select\{\mathbf{B}\} &= trapmf(Y : 0.1, 0.3, 0.3, 0.5) \\ Select\{\mathbf{M}\} &= trapmf(Y : 0.3, 0.5, 0.5, 0.7) \\ Select\{\mathbf{G}\} &= trapmf(Y : 0.5, 0.7, 0.7, 0.9) \\ Select\{\mathbf{VG}\} &= trapmf(Y : 0.7, 1, 1, 1). \end{aligned}$$

2.3.2 Design a Fuzzy Rule Base

Table 3 is Brante's $3 * 3$ relationship matrix of input and output. Therefore, the rule bases will get 9 fuzzy rules.

Table 3: The relation matrix of Brante's fuzzy rule

Re\CSI	**W**	**A**	**S**
L	VB	B	B
M	B	M	G
F	M	G	VG

2.3.3 Fuzzification and Defuzzification

The Aggregation Method of Brante's fuzzy systems used Max and Centroid for defuzzification. As the general, a fuzzy systems often output a same value when defuzzification results in the same value. A random selection method is used to determine a relay node by Brante's algorithm. Brante's method get a better simulation results when comparison with a traditional random selection relay algorithm and an opportunity chosen relay algorithm, which also play a simple calculation mechanism of fuzzy systems and extend the network life cycle.

Yang and Brante officially used the fuzzy theory for relay selection algorithm in wireless sensor networks. This paper use the same framework to resolve its methodology. Comparing their architecture with a typical fuzzy system, the difference is shown in Table 4.

3 Safety Testing Systems

This section describes a relay selection algorithms based on fuzzy systems in wireless sensor networks to evaluate when it suffered an attack in an unsafe environment. Some literature points out that the biggest threat in the battery-powered wireless sensor networks are Sinkhole Attacks, Wormhole Attacks and Sybil Attacks. It directly affects the algorithms results to regulate power consumption and communication quality [22]. A common threat in attacks mode is power consumption [15]. It's often increasing the burden of relaying nodes, when a sending packet was dropped by an attacked node. Therefore, the adjusting mechanism of algorithm will lose balance. Then it can't reach the expected purpose to extend the network life cycle.

Since Yang's and Brante's proposed methods are assumed to be performed in an unattacked state. In fact, the effectiveness of the security is different between each designed fuzzy-based inference rules system, when it suffers a power attack.

Considering this scenario, the attacker increase antenna power of attack nodes , and expand its transmission range, or forge remaining power status or otherwise supplemental power of attack nodes, so that adjacent nodes mistake it reliable cooperative node, and then select the attack node as a relay node.

Therefore, if an algorithm doesn't consider these cases, the mechanism of selection algorithm can easily be exploited by attackers. It has consumed too much power without any awareness. In order to evaluate efficacy and safety of different relay selection algorithms designed by fuzzy-based inference rules, this study design a security testing systems, and assess the method of construction mode with fuzzy-based rules base in wireless sensor network. It is described as follows:

The security testing System includes five components: network architecture, signal processing, power consumption, security assessment of simulated attacks and simulation platform.

3.1 Network Architecture

As the study points out Brante's method has relatively good performance. Therefore, this study used Brante's architecture in networks [4] as shown in Figure 4 and Table 5.

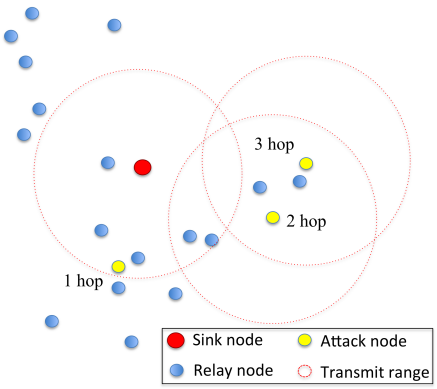

Figure 4: Simulation topology in wireless sensor networks

3.2 Signal Processing

The signal processing follows the IEEE 802.15.4 specification, and use the data transfer mode [9] in networks. Mixing with white noise processing in simulation [14, 19], it assumes that the receiver sensitivity is -85dBm, and the maximum transmission distance is 200M.

3.3 Power Consumption

The power consumption is using the first order radio model radio power consumption model [8, 13] as shown in Figure 5 and Table 6, and ignoring the tiny fixed detection power calculation factor on electricity consumption [3, 7].

Table 4: Comparing the architecture method with Yang's and Brante's fuzzy system

Fuzzy Systems Construction	Yang's system	Brante's system
Variables domain range	$Re = [0 \sim 100\%]$ $CSI = [-255dBm \sim 0dBm]$	$Re = [0 \sim 100\%]$ $CSI = [-255dBm \sim 0dBm]$
Variable parameters	2 inputs are all divided into 1 set undefined output	2 inputs are all divided into 3 sets 1 output is divided into 5 sets
Membership function	trimf	trapmf
Fuzzy rules base	Undefined	9 sets
Defuzzification	Traditional	Centroid

Table 5: Simulation parameters in networks

Network topology	Mesh topology
Node numbers	20 nodes (including Sink)
Transmission distance	200 M
Physical layer MAC protocol	802.15.4
Maximum number of network layers hops	3 layers
Data transfer mode	Beacons

Table 6: Power consumption in model

Operational unit	**Power consumption**
Transmitter electronics (E_{Tx_elec}) **Receiver electronics** (E_{Rx_elec}) E_{Tx_elec}=E_{Rx_elec}=E_{elec}	$50nJ/bit$
Transmission amplifier (E_{amp})	100 $pJ/bit/m^2$

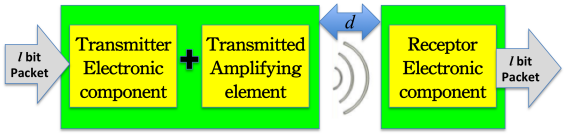

Figure 5: Radio power consumption model

3.4 Simulated Attack and Safety Assessment

The safety effectiveness of relay selection algorithm with fuzzy-based inference system was assumed that environment status were unprotected or lost protection in network, and its defensive system didn't even start yet. An attacked node set its each positions in 3 hop layers in Figure 4, separately performing four different attack patterns to construct the evaluation of system modules as shown in Table 7.

The simulation was set under the same network topology. All of 20 nodes starts attack after 50 rounds and ends attack after at 200 rounds. An attacked number was counted by relay nodes in selection algorithm. And the safety rate was indicated as the security level of algorithms by conversion as follows:

$$Safety\ Ratio = \frac{candidate\ attack\ numbers - attacked\ numbers}{candidate\ attack\ numbers} \times 100\%.$$

Therefore, the lower total number of attacks were the better performance of hedging function, whereas the higher total number of attacks were the worst performance of hedging function. In contrast, an algorithm with the better performance of hedging function able to extend the network life, because they can avoid the attacked nodes and without waste retransmission for failure relay. Thus, an algorithms can achieve its purposes for balancing power consumption and communication quality.

3.5 Simulation Platform

The simulation platform used the MATLAB Design and simulate fuzzy logic systems [10]. It can execute two kinds fuzzification connection methods of And-Min or Or-Max and five kinds defuzzification approachs about Centroid (center of gravity method), Bisector (the qualitative method), Lom (maximum membership degree method), Mom (middle membership degree method), and Som (minimum membership degree method). As long as the

Table 7: Consumption attack type with *CSI* and *RE*

Type	Patterns and metrics (*CSI* & *Re*)	
1	Signal attack patterns	The attacked nodes utilize the enhanced power mode to change the signal value of attacked nodes in select-right nodes
2	Power attack patterns	The attacked nodes utilize the enhanced energy mode to change the remaining energy value of attacked nodes in right selected nodes
3	Hybrid attack patterns	The attacked nodes utilize the enhanced the power and energy mode to change the signal and remaining energy value of attacked nodes in right selected nodes
4	Uncooperative attack patterns	The attacked nodes themselves don't change the power and energy mode, and don't change the signal and the remaining energy value of attacked nodes in right selected nodes. It plays the role of uncooperative and doesn't reply the data to receive the information

testing system was replaced other fuzzy inference module. Any relay selection algorithm of fuzzy systems can operate in this testing system. The testing system will store all candidate nodes information in different tests of attack patterns. A single fuzzy system was displayed in face different attack patterns from the real work. There are nearly got ten thousand of records. To aggregated these cumulative test data's. Thats some valuable experience for fuzzy system to analysis the safety of relay selection algorithms. Because the rules base of fuzzy inference system is very dependent on experience of expert and practical operation [26]. Extracting of those experiences will be transformed into fuzzy-base algorithm to improve its safety policy in the future work.

4 The Safety Test Results of Existing Fuzzy Relay Selection Algorithms

This section describes the test results of Yang's and Brante's fuzzy-based relay selection algorithms in wireless sensor networks when they faces three attack positions and four attack patterns in the study testing system.

Figure 6 shows the overall performance of the existing algorithms of Yang and Brante. First, the 2hop safety rate is always higher the 1hop when the position was launched by the attacker. The outer layer is higher by 4% than the inner layer of safety rate. It's higher by 3.8% of Brante's algorithm due to the reason of the relationship between network topology. The inner layer of the network transmission load will come naturally heavier than the outer layer. That means the inner network attackers have obtained the highest efficiency, too. Further, the 3hop at outermost has 100% safety rate due to the network environment factors. There is no difference between Yang's and Brante's algorithms. This also explains the outermost layer of the network, so it doesn't need to pass as a relay node. Therefore, the risk is close to zero.

For hedging effect view of the algorithms, Brante's algorithm is higher than 0.9% of Yang's algorithm under 1hop attack. It's also higher 0.7% under 2hop attack.

Therefore, hedge effectiveness of test results of Brante simulation algorithm is better 0.8% better than Yang's algorithm in average.

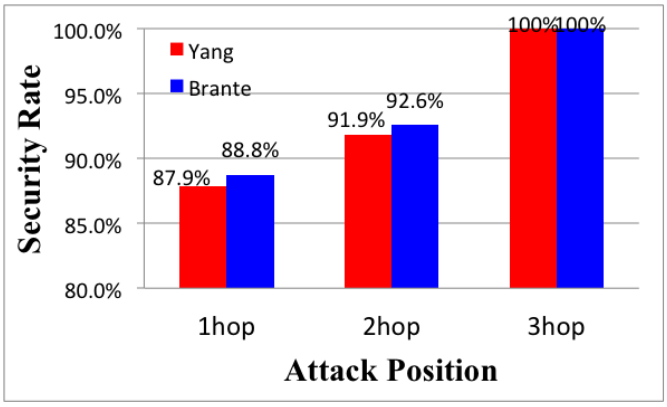

Figure 6: Existing algorithms, Attack Position, hedge effectiveness

Figures 7 and 8 show Yang's and Brante's algorithms facing four kinds of attack patterns to hedge effectiveness under attack in 1hop position and 2hop position. Those relay selection algorithms are concerned with the same fuzzy inference developed in wireless sensor networks, and whose defense strength order was: Type4 (*Dicard*) > Type2 (*Re*) > Type1 (*CSI*) > Type3 (*CSI* + *Re*), when Yang's and Brante's algorithms encounter four kinds of attack patterns. That explains the Type3 attack patterns are the most difficult to guard against by mixed signal and power. The next was the Type1 attack patterns by the signal, next was the Type2 power attack patterns, and, finally, the Type4 attack patterns are not so much affected by uncooperative loss threat.

When viewed in Figures 7 and 8 at same time, it shows the hedge effectiveness in Yang's and Brante's algorithms. It's more difficult to distinguish the difference between algorithms when it is more to outer layer of the network. This phenomenon can be applied to all the fuzzy-based relay selection algorithms for the safety assessment in wire-

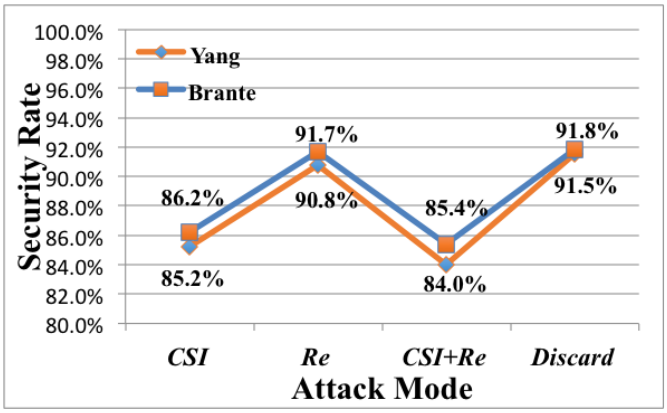

Figure 7: Existing algorithms, Attack Mode by 1 hop Position, hedge effectiveness

less sensor networks.

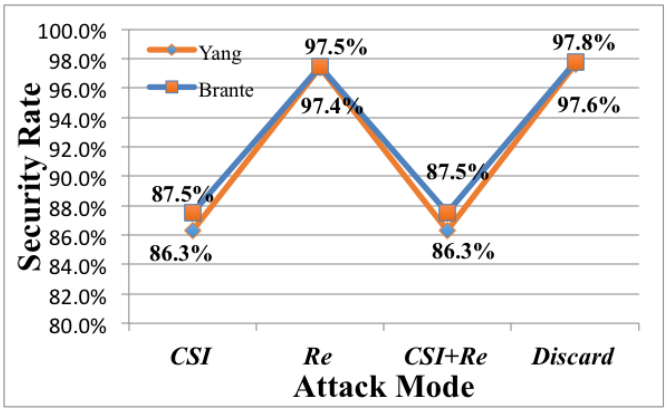

Figure 8: Existing algorithms, Attack Mode by 2hop Position, hedge effectiveness

5 Future Research

The fuzzy theory shows good performance. It was originally regarded as the best of the tools in the household appliance and automatic control. But most scholars generally believe that fuzzy is not productive to safety. Luckily, Yang and Brante integrated the fuzzy theory with relay selection algorithms in wireless sensor networks. And they gained considerable achievements, so they pioneered out the research-oriented. This section will present a number of research directions to let interested scholars sustain research results in this field.

5.1 Fuzzyfication and Defuzzyfication Methodology

Because Yang and Brante used different fuzzyfication and defuzzification methods (Table 8). Those show different performance results on the simulation testing. In the methodology, Table 9 showed output results by the typical architecture of variable parameter rule base in Figure 2. There are two kinds connection methods, And-Min and Or-Max, for input and output, and five kinds defuzzification methods, Centroid, Bisector, Lom, Mom and Som, between the inference process; perhaps the value order was the same, but after the inference, the results would have been completely different because they used differences methodology. When they choose on doctrinal situation which get interaction between fuzzy membership functions and fuzzy rules base, the calculation results are very different from the traditional linear algorithm.

Therefore, we study the difference hedge effectiveness in the attack state when they used different fuzzification and defuzzification methods in the same domain [16] to find or apply existing algorithms, thereby obtaining more suitable methodology for fuzzy inference system. Even to import C-Mean clustering or variable domain range in fuzzy theory [20]. The formation of strategic inference rules will contribute to the development of high safety relay selection algorithms in wireless sensor networks.

5.2 Fuzzy Rules Base

The core value of the fuzzy rules base is to translate a control rule of linguistic with "If X1 and X2 Then Y" based on experience and expert knowledge. Under this criterion, the relationship between inputs and outputs all rely on control rules. Usually, if the input is divided into three fuzzy parameters, there will be deduced nine fuzzy rules from 3 * 3 matrix of input and output relationship. How detailed should the rule base be controlled? The affect was complete limited by the number of input variables and the division of output number. Therefore, the study of membership function can focus on these numbers to make the difference and develop specific rule base of relay selection algorithm in wireless sensor networks which accumulate sufficient data pass through the relay from simulation. After all, the fuzzy rule base is not imagined, it needs to have the support of big data in order to derive a practical and effective experience and knowledge.

Table 8: Academics using fuzzification and defuzzification

Methods √ Used × Unused ∼ Similar		**Yang**	**Brante**
Fuzzification Connection	And-Min	×	√
	Or-Max	×	×
Defuzzification	Centroid	×	√
	Bisector	×	×
	Lom	×	×
	Mom	∼	×
	Som	×	×

Table 9: Example of fuzzy inference rule base

Input		Connection	Defuzzy	Output	Defuzzy shown
Re	*CSI*				
60 %	-75 dBm	And-Min	Centroid	0.706	
			Bisector	0.73	
			Lom	0.95	
			Mom	0.75	
			Som	0.65	
		Or-Max	Centroid	0.545	
			Bisector	0.55	
			Lom	1	
			Mom	0.594	
			Som	0.2	

5.3 Fuzzy Feedback Control System Parameters

Figure 9 is a diagram of fuzzy-bases feedback control system [27]. The purpose of the control system is to maintain the y_P output within the range of the SP set point. The controller is adjusted with the e of deviation and the Δe of change deviation as follows:

$$\begin{aligned} e(n) &= SP(n) - y_p(n) \\ \Delta e &= e(n) - e(n-1) = -(y_p(n) - y_p(n-1)). \end{aligned}$$

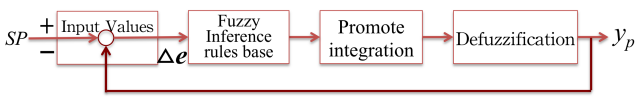

Figure 9: Basic fuzzy feedback control system diagram

The study on fuzzy feedback mechanism could control a relay selection algorithm to maintain security within a certain degree. Then it's unlikely to bog down under the attack and obtain stop-loss effect.

6 Conclusion

The relay process is a noteworthy topic about safety of relay selection algorithm in an unattended sensing network environment. This paper explored the use of fuzzy theory in a relay selection algorithm framework and focused on existing research to balance between communication quality and consuming energy in a wireless sensor network. In order to consider the safety for a fuzzy-based relay selection algorithm, safety testing and evaluation systems were also constructed in wireless sensor networks to analyze the safety effectiveness of existing algorithms about Yang and Brante. In future studies, this paper proposes three main directions. It mainly hopes to make more attention and progress study based on a fuzzy theory in relay selection algorithm. It will improve the research results to follow this type of rely selection algorithms out from prevention systems and protection systems.

References

[1] S. Abdulhadi, M. Jaseemuddin, A. Anpalagan, "A survey of distributed relay selection schemes in cooperative wireless ad hoc networks," *Wireless Personal Communications*, vol. 63, no. 4, pp. 917–935, 2012.

[2] I. F. Akyildiz, Y. S. W. Su, and E. Cayirci, "Wireless sensor networks: A survey," *Computer Network*, vol. 38, no. 4, p. 30, 2002.

[3] M. Asadi, C. Zimmerman, and A. Agah, "A game theoretic approach to security and power conservation in wireless sensor networks," *International Journal of Network Security*, vol. 15, no. 1, pp. 50–58, 2013.

[4] G. Brante, G. d. S. Peron, and et al. R. D. Souza, "Distributed fuzzy logic-based relay selection algorithm for cooperative wireless sensor networks," *IEEE Sensors Journal*, vol. 13, no. 11, pp. 4375–4386, 2013.

[5] T. M. Cover and A. A. E. Gamal, "Capacity theorems for the relay channel," *IEEE Transactions Information Thbory*, vol. 25, no. 5, pp. 572–584, 1979.

[6] S. Faye and J. F. Myoupo, "Secure and energy-efficient geocast protocol for wireless sensor networks based on a hierarchical clustered structure," *International Journal of Network Security*, vol. 15, no. 3, pp. 151–160, 2013.

[7] M. N. Halgamuge, M. Zukerman, and K. Ramamohanarao, "An estimation of sensor energy consumption," *Progress in Electromagnetics Research B*, vol. 12, pp. 259–295, 2009.

[8] W. R. Heinzelman, A. Chandrakasan, and H. Balakrishnan, "Energy-efficient communication protocol for wireless microsensor networks," in *Proceedings of the 33rd Hawaii International Conference on System Sciences*, pp. 1–10, 2000.

[9] IEEE Standards Association, "IEEE 802.15.4 standards," 2011.

[10] J.-S. Roger Jang, N. Gulley, "Fuzzy logic toolbox user's guide," 2015. (http://andrei.clubcisco.ro/cursuri/5master/ciblf/Artificial_Intelligence_-_Fuzzy_Logic_Matlab.pdf)

[11] V. Katiyar and N. Chand, "Recent advances and future trends in wireless sensor networks," *International Journal of Applied Engineering Research*, vol. 1, no. 3, pp. 330–342, 2010.

[12] B. Krishnamachari, *Networking Wireless Sensors*, Cambridge University Press, 2005.

[13] T. Landstra, S. Jagannathan, and M. Zawodniok, "Energy-efficient hybrid key management protocol for wireless sensor networks," *International Journal of Network Security*, vol. 9, no. 2, pp. 121–134, 2009.

[14] D. U. Lee, W. Luk, and et al. J. D. Villasenor, "A gaussian noise generator for hardware-based simulations," *IEEE Transactions on Computers*, vol. 53, no. 12, pp. 1523–1534, 2004.

[15] W. T. Li, T.-H. Feng, and M.-S. Hwang, "Distributed detecting node replication attacks in wireless sensor

networks: A survey," *International Journal of Network Security*, vol. 16, no. 5, pp. 323–330, 2014.
[16] M. Maeda and S. Murakami, "A self-tuning fuzzy controller," *Fuzzy Sets and System*, vol. 51, pp. 29–40, 1992.
[17] E. H. Mamdani, "Application of fuzzy logic to approximate reasoning using linguistic synthesis," *IEEE Transactions on Computers*, vol. C-26, no. 12, pp. 1182–1191, 1977.
[18] E. H. Mamdani and S. Assilian, "An experiment in linguistic synthesis with a fuzzy logic controller," *International Journal of Man-Machine Studies*, vol. 7, no. 1, pp. 1–13, 1975.
[19] Obata and Nobuaki, *White Noise Calculus and Fock Space*, Berlin, Springer, 1994.
[20] W. Pedrycz, "Fuzzy sets in pattern recognition: Methodology and methods," *Pattern Recognition*, vol. 23, no. 1-2, pp. 121–146, 1990.
[21] W. Pedrycz and F. Gomide, *An Introduction to Fuzzy Sets: Analysis and Design*, Cambridge MA, USA: MIT Press, 1998.
[22] S. Rekhis and N. Boudriga, "Pattern-based digital investigation of x-hole attacks in wireless adhoc and sensor networks," in *Ultra Modern Telecommunications & Workshops (ICUMT'09)*, pp. 1–8, 2009.
[23] V. K. Shah and A. P. Gharge, "A review on relay selection techniques in cooperative communication," *International Journal of Engineering and Innovative Technology*, vol. 2, no. 5, pp. 65–69, 2012.
[24] H. S. Soliman and M. Omari, "Application of synchronous dynamic encryption system (sdes) in wireless sensor networks," *International Journal of Network Security*, vol. 3, no. 2, pp. 160–171, 2006.
[25] M. Sugen, *Industrial Applications of Fuzzy Control*, New York, NY, USA: Elsevier Science Inc, 1985.
[26] M. Sugeno, "Fuzzy measures and fuzzy integrals: a survey," *Fuzzy Automata and Decision Processes*, vol. 78.33, pp. 89–102, 1977.
[27] R. R. Yager and D. P. Filev, *Essentials of Fuzzy Modeling and Control*, New York: Wiley, 1994.
[28] W. Yang, Y. Cai, and Y. Xu, "An energy-aware relay selection algorithm based on fuzzy comprehensive evaluation," in *Proceedings of International Conference on Network Security, Wireless Communication Trusted Computing*, pp. 144–146, 2009.
[29] L. A. Zadeh, "Fuzzy sets," *Information and Control*, vol. 8, no. 3, pp. 338–353, 1965.
[30] L. A. Zadeh, "The concept of a linguistic variable and its application to approximate reasoning," *Learning Systems and Intelligent Robots*, pp. 1–10, 1974.
[31] C. Zhai and J. Liu and L. Zheng, et al., "Maximise life-time of wireless sensor networks via a distributed cooperative routing algorithm," *Transactions on Emerging Telecommunications Technologies*, vol. 23, no. 5, pp. 414–428, 2011.

Tung-Huang Feng received his M.S. in Information Management from Chao-Yang University of Technology, Taichung, Taiwan, ROC, in 2002. He is currently pursuing the Ph.D. degree from Computer Science Information Engineering Department of Asia University, Wufeng, Taiwan. His research interests include information security, cloud computing, networked control system and Sensor Networks.

Neng-Yih Shih was born in Changhua, Taiwan, in 1959. He received his B.S and M.S. degrees in the Department of Automatic Control Engineering, from Feng Chia University, Taiwan, in 1982 and 1984. He received a Ph.D. degree in Institute of Aeronautics and Astronautics from the Nation Cheng Kung University, Taiwan, in 2001. He is currently an associate professor with the Department of Computer Science and Information Engineering, Asia University, Taiwan. He is also Secretary General of Asia University, a position he has held since Aug. 2011. His main research interests include the application of expert control, networked control system and intelligent systems.

Min-Shiang Hwang received the B.S. in Electronic Engineering from National Taipei Institute of Technology, Taipei, Taiwan, Republic of China (ROC), in 1980; the M.S. in Industrial Engineering from National Tsing Hua University, Taiwan, in 1988; and the Ph.D. in Computer and Information Science from National Chiao Tung University, Taiwan, in 1995. He also studied Applied Mathe- matics at National Cheng Kung University, Taiwan, from 1984-1986. Dr. Hwang passed the National Higher Examination in field Electronic Engineer. in 1988. He also passed the National Telecommunication Special Examination in field Information Engineering, qualified as advanced technician the first class in 1990. From 1988 to 1991, he was the leader of the Computer Center at Telecommunication Laboratories (TL), Ministry of Transportation and Communications, ROC. He was also a chairman of the Department of Information Management, Chaoyang University of Technology (CYUT), Taiwan, during 1999-2002. He was a professor and chairman of the Graduate Institute of Networking and Communications, CYUT, during 2002-2003. He is currently a professor of the department of Management Information System, National Chung Hsing University, Taiwan, ROC. He obtained 1997, 1998, 1999, 2000, and 2001 Outstanding Research Awards of the National Science Council of the Republic of China. He is a member of IEEE, ACM, and Chinese Information Security Association. His current research interests include electronic commerce, database and data security, cryptography, image compression, and mobile computing. Dr. Hwang had published 170+ articles on the above research fields in international journals.

Hiding of Confidential Data in Spatial Domain Images using Image Interpolation

S. Maria Celestin Vigila[1] and K. Muneeswaran[2]
(Corresponding author: S. Maria Celestin Vigila)
Department of Information Technology, Noorul Islam University[1]
Kumaracoil - 629 180, Tamilnadu, India
Department of Computer Science and Engineering, Mepco Schlenk Engineering College[2]
Sivakasi - 626 005, Tamilnadu, India
(Email:{celesleon, kmuni12}@yahoo.com)
(Received May 28, 2013; revised and accepted Feb. 10 & May 20, 2014)

Abstract

Data hiding is a technique that is used to embed secret data into a cover media. This paper presents the implementation of reversible data hiding in spatial domain images based on neighbor mean image interpolation without impairing the image quality. Here, the cover image and secret bits are extracted from stego image without the need for any additional information. The strength of the proposed method has lower computational complexity, less blurring and greater image resolution. This paper also discusses the performance aspects of the proposed method which is superior in terms of high data embedding capacity and image quality.

Keywords: Image interpolation, payload, reversible data hiding

1 Introduction

In the modern era of high speed internet, multimedia data are represented in digital forms to be transmitted through internet. Since digital media is easily replicated and subject to tampering, protecting content is an important issue. Data hiding schemes have been widely used to protect the content of digital media. Data hiding schemes generally embed important data into the cover media by modifying the pixels to protect the data from illegal peeking or damaging. The concept of data hiding was first suggested by Simmons in 1983 [11]. Data can be hidden in lots of ways. In order to hide secret data, straight data insertion may encode every bit of data in the cover media or it may selectively embed data in noisy areas that describe a smaller amount of attention. Data may also be scattered erratically throughout the cover media. There are a number of ways to hide secret data; the most common methods are the Least Significant Bit (LSB) insertion, masking, filtering and transformations [7].

Reversible data hiding methods enable the exact recovery of the original cover image from the stego image without any distortion upon extraction of the embedded information [6]. Image interpolation tackles the difficulty of generating high resolution images from its low resolution image. The model that is employed to illustrate the relationship between high resolution and low resolution pixels plays a crucial role in the performance of an interpolation algorithm. Conventional linear interpolation methods, derived from space-invariant models, are unsuccessful to capture the hasty evolving statistics around edges and annoying artifacts. Linear interpolation [8] is commonly chosen for computational simplicity not for performance. Hence, reversible data hiding is incredibly important for securing sensitive data in image applications.

With this goal, the implementation of reversible data hiding in spatial domain images based on neighbor mean image interpolation without impairing the image quality is presented in this paper. The cover image and secret text data are extracted from stego image without the need for any additional information. The proposed method can embed a huge amount of secret data whereas maintaining an elevated image quality that of the other data hiding methods.

This paper is organized as follows. Section 2 reviews related works in the area of data hiding. In Section 3, the details of the proposed data embedding method is described. In Section 4, the experimental results are presented and discussed. Finally, the concluding remarks are presented in Section 5.

2 Related Works

In this section, some highlights of the relevant work in the area of data hiding are outlined. A simple and most

instinctive scheme for hiding secret data into a digital image is to directly substitute the LSBs of each pixel in the cover image with the bits of secret data [17]. Wang et al. [16] described an optimal re-naming problem for the hidden secret data, and then applied a genetic algorithm for seeking the problem's nearly optimal solution. Thien and Lin [13] suggested a digit by digit data hiding scheme based on modulus function.

Ni et al. [9] proposed a histogram based data hiding scheme. In their scheme, the search for the pair of peak and zero points from the histogram is performed first. The peak point refers to the most often occurring pixel value in the histogram. The zero point stands for the pixel value with zero or minimal occurrences in the histogram. The secret data are embedded by shifting the pixel values located between the peak point and the zero point. Huang et al. [5] proposed a histogram-based scheme which used a multilevel hiding strategy to obtain high capacity and low distortion.

Most of the work on reversible data hiding focuses on the data embedding/extracting on the plain spatial domain [1, 3, 4, 14]. Tsai et al. [15] developed a reversible data embedding method that merges predictive coding and histogram shifting techniques. Chang et al. [2] suggested a reversible data embedding method to embed secret data in original images based on the edge directed prediction scheme. In this scheme, an embedded pixel value is simplifying along with a predetermined threshold and the difference between the predicted pixel value and its original pixel value.

Peng et al. [10] proposed a reversible data embedding algorithm based on integer transform and adaptive embedding. In this algorithm, the parameter is adaptively chosen in dissimilar blocks in integer transform. In [12], Tai et al. presented a reversible data hiding scheme based on histogram modification using pairs of peak points. Owing to these existing works on data hiding and its popularity, it is proposed to hide the data in spatial domain images using neighbor mean image interpolation.

3 Proposed Method

The proposed method is made up of interpolation, data embedding and data extraction phases. Initially, the proposed method introduces a scaling up and an interpolation technique. The scaling up image focuses on high speed and low complexity that is used as a cover image. In the data embedding phase, the secret data S is taken as input and then, it converts into binary secret bits. After the conversion process, then these secret bits are embedded into the cover image C, and then transmit the stego image to a receiver without impairing the image quality.

At receiving side, the cover image and secret bits are extracted from stego image without the need for any additional information. Then these secret bits are converted into text data. Figure 1 shows the sketch of the proposed data embedding method.

3.1 Interpolation Phase

In the Neighbor Mean Interpolation (NMI) phase uses neighboring pixel values to compute the mean, and after that the computed mean value is introduced into a pixel that has not been assigned yet. In general, high resolution pixels are getting when a neighboring pixel values are referenced in order to calculate a value that is to be assigned, but time complexity is higher when the number of referenced pixel is higher. The scaling up method decides what application to which it should be applied.

$$p'(x,y) = \begin{cases} p(x,y) & if\ x = k.i, y = k.j \\ [p(x,y-1) + p(x,y+1)]/k & \\ & if\ x = k.i, y = k.j+1 \\ [p(x-1,y) + (p'(x+1,y)]/k & \\ & if\ x = k.i+1, y = k.j \\ [p(x-1,y-1) + p'(x-1,y) + & \\ \quad p'(x,y-1)]/k+1) & otherwise \end{cases} \quad (1)$$

Assume that $p(x,y)$ represents the value of a pixel located at (x,y) in original image and its interpolated pixel in cover image $p'(x,y)$ is computed as Equation (1), where $0 \le y \le x$ and $i,\ j = 0, 1, \cdots, 127$. k stands for a value of scaling up coefficient. In the scaling up process, the cover image is scaled two times more. Therefore, the value of k is 2 for which a cover image can preserve an elevated resolution.

For the pixel $p'(0,0)$ and $p'(2,2)$ are the same value with $p(0,0)$ and $p(2,2)$, respectively. In the case of $x < y$, $p'(0,1)$ is computed as $(p(0,0) + p(0,2))/2$ operation. When $x < y$, $p'(1,0)$ is calculated as $(p(0,0) + p(2,0))/2$. Finally, $p'(1,1)$ is obtained from $(p(0,0) + p'(0,1) + p'(1,0))/3$.

The proposed neighbor mean interpolation method is more efficient, less blurring and greater image resolution than nearest neighbor interpolation and bilinear interpolation methods. So, the resulting image that is produced by the NMI method is used with a cover image.

3.2 Data Embedding Phase

In the data embedding phase, the sequence of data embedding can be in zig-zag, left-to-right and top to bottom directions. Before secret data are embedded, the cover image is partitioned into four-pixel, non-overlapping, consecutive by zig-zag scanning. For every four non-overlapping consecutive pixel values, i.e., $p(x,y)$, $p(x,y+1)$, $p(x+1,y)$ and $p(x+1,y+1)$, the corresponding stego-image pixel values are $p'(x,y)$, $p'(x,y+1)$, $p'(x+1,y)$, and $p'(x+1,y+1)$, respectively. Here, secret data are embedded into three pixels except for $p(x,y)$ pixel. The details of each step are specified as follows.

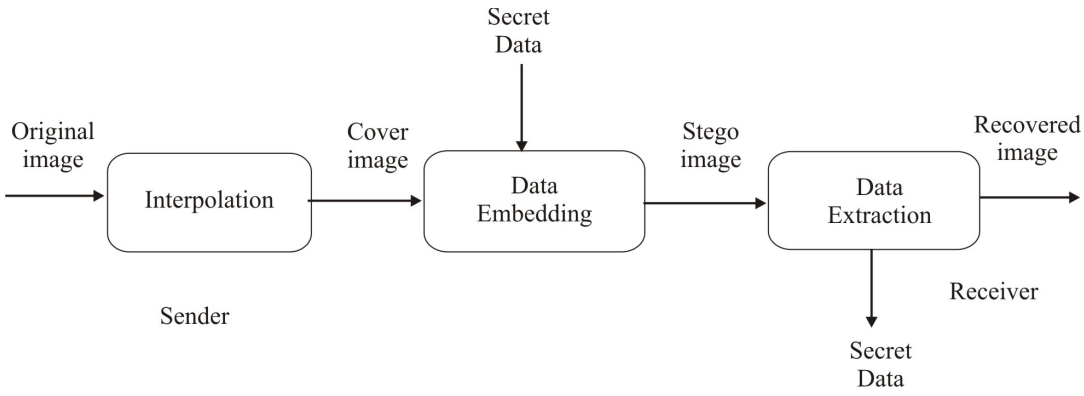

Figure 1: Sketch of proposed data hiding method

1) For every four non-overlapping consecutive pixel values, a difference value d is computed as Equation (2), where $0 \leq x, y \leq 127$ and β, α value is 0 or 1, respectively. When $\beta = \alpha = 0$, the difference value d as zero.

$$d = p(k.x + \beta, k.y + \alpha) - p(k.x, k.y). \quad (2)$$

2) Calculate the number of bits, say n, which can be embedded in this pixel is

$$n = \log_2 |d|. \quad (3)$$

3) Select first n bits from the secret message S that is substream $S(n)$ and it is converted to integer value b.

4) Then, a stego image pixel $p'(x, y)$ is computed as follows.

$$p'(x, y) = p(x, y) + b. \quad (4)$$

50	83	116	50	102	116
132	88	105	137	95	105
214	154	94	214	154	94

a) Cover Image b) Stego Image

Figure 2: Example results of the data embedding procedure

The following serves as a detailed example to depict the data embedding procedure. Figure 2(a) shows a 3∗3 cover image, and Figure 2(b) shows its stego image. Let secret message $S =$ "1001100010100111". In Figure 2(a), pixel $p(0, 0) = 50$ is the starting point for zig-zag scanning. In the proposed method, pixel $p(0, 0)$ is retained. Next, consider the pixel $p(0, 1)$ as 83. First the difference value d is computed by using Equation (2) as $d = 83 - 50 = 33$. Second, by Equation (3) calculate the number of bits embedded in $p(0, 1)$ as $\log_2 \mid d \mid = 5$. Third, select first 5 bits from the secret message S and convert it into integer value $b = 10011_2 = 19$. Therefore, the pixel of stego-image is $p'(0, 1) = 83 + 19 = 102$. Similarly, other pixel values $p(1, 0)$ and $p(1, 1)$ are calculated and tabulated in Table 1.

Table 1: Example details of the data embedding procedure

Coordinate	$p(x, y)$	d	n	S(n)	b	$p'(x, y)$
(0, 1)	83	33	5	10011	19	102
(1, 0)	132	82	6	000101	5	137
(1, 1)	88	38	5	00111	7	95

3.3 Data Extraction Phase

In the data extraction phase, the cover image and secret bits are obtained by using stego image only without need for any further information. The cover image and the secret data are extracted by using the following step by step procedure.

1) Compute the pixel in cover image $p(x, y)$ from the stego image $p'(x, y)$ using a simple arithmetic expression as defined by Equation (5), where $0 \leq y \leq x$ and $i, j = 0, 1, \cdots, 127$ and k is defined to 2.

$$p(x,y) = \begin{cases} [p'(x,y) + p'(x,y)] \\ \qquad if\ x = k.i, y = k.j \\ [p'(x,y) + p'(x,y+1)]/k \\ \qquad if\ x = k.i, y = k.j + 1 \\ [p'(x,y) + (p'(x+1,y)]/k \\ \qquad if\ x = k.i + 1, y = k.j \\ [k.p'(x,y) + p'(x,y+2)/k + \\ \quad p'(x+2,y)/k]/(k+1) \\ \qquad otherwise \end{cases} \quad (5)$$

2) Then, the secret data embedding in $p(x, y)$ is computed as

$$b = p'(x, y) - p(x, y). \quad (6)$$

3) Next, calculate the difference value d using Equation (2) for the three neighboring pixels, excluding $p(0, 0)$; this is because $p'(0, 0)$ does not include secret bits.

4) Finally, the length of hiding is calculated for each secret bit. The integer value b is represented as secret bits based on the hiding length.

Table 2: Example details of the data extraction procedure

Coordinate	$p'(x, y)$	$p(x, y)$	b	d	n	S(n)
(0,1)	102	83	19	33	5	10011
(1,0)	137	132	5	82	6	101
(1,1)	95	88	7	38	5	111

Table 2 shows an example of the data extraction procedure. For example, $p(0, 1)$ is calculated by $(p'(0, 0)+ p'(0, 2))/2 = (50 + 116)/2 = 83$ and the secret data embedding in $p(0, 1)$ is $102 - 83 = 19$. $p(1, 0)$ is also calculated by $(p'(0, 0) + p'(2, 0))/2 = (50 + 214)/2 = 132$ and the secret data embedding in $p(1, 0)$ is $137-132 = 5$. Finally, $p(1, 1)$ is obtained by $[2 * p'(0, 0) + p'(0, 2)/2 + p'(2, 0)/2]/3 = (2 * 50 + 116/2 + 214/2)/3 = 265/3 = 88$ and secret data embedding in $p'(1, 1)$ is $95-88 = 7$. Then calculate the difference value d as 33, 82 and 38 which correspond to $p'(0, 1)$, $p'(1, 0)$ and $p'(1, 1)$. Finally, based on the hiding length n, the integer value b is represented as secret bits as $19 = 10011_2$, $5 = 000101_2$ and $7 = 00111_2$. As a result, the cover image and secret data are extracted from the stego image only.

4 Results and Discussion

In this paper, four standard 512×512 gray scale images Lena, Pepper, Airplane and Baboon are considered as cover image for data embedding phase as shown in Figure 3. To evaluate the performances of the proposed method, two indicators have been considered.

Foremost, the histogram of the cover image and stego image are compared. An image histogram shows how pixels in an image are distributed by plotting the number of pixels at each intensity level. The histogram of the stego image is slightly varies from its cover image. So, the image distortion gets reduced in this proposed method. Due to this reason, the hacker cannot find the hidden data, because stego image looks same as original image but the secret data is embedded in stego image. From this, the proposed method is more efficient to hide the secret data by verifies the histogram of the original and stego images as shown in Figure 4.

Second, the Peak Signal-to-Noise Ratio (PSNR) is evaluated to compare visual quality between the cover image and the stego-image. The PSNR value can be computed by the following equation:

$$PSNR(dB) = 10 \times \log \frac{(2^n - 1)^2}{MSE}. \quad (7)$$

Where n represents the number of bits per pixel, and MSE (Mean Square Error) can be computed as follows:

$$MSE = \frac{1}{u \times v} \sum_{0}^{u-1} \sum_{0}^{v-1} (E_{xy} - C_{xy})^2. \quad (8)$$

The parameters u and v represent the height and width of the image. The notations C_{xy} and E_{xy} represent the cover image and the stego image pixel value in position (x, y) respectively. If the distortion between the cover image and the stego image is small, the PSNR value is large. Thus, a larger PSNR value means that the quality of the stego image is better than the smaller one. Generally, with human vision alone, it is hard to differentiate a stego image from its original image when the PSNR value is greater than 30 dB.

Table 3 gives the values of PSNR calculated for the four cover images with different payload size are stated by the formula specified in Equation (7).

Figure 3: The Four Cover Images

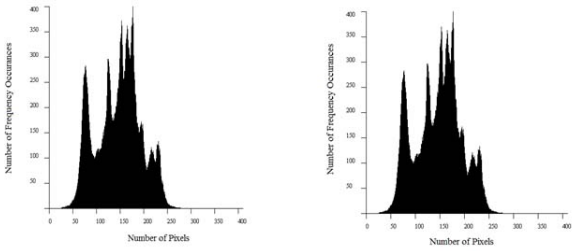

Figure 4: Histogram of original image and stego image

Here we include the results for the 512×512, 8 bits per pixel (bpp) grayscale "Lena". The embedded payload size

Table 3: PSNR values for the four cover images with different payload size

Cover image	Payload Size (bits)					
(512 x 512)	2520	36216	42264	56840	71400	85960
Lena	66.7	61.93	54.19	51.56	49.94	48.76
Peppers	55.29	54.44	53.23	51.02	49.55	48.46
Airplane	63.28	59.54	52.82	50.76	49.37	48.32
Baboon	56.96	55.76	51.7	50.03	48.83	47.89

and its PSNR of embedded "Lena" images are specified in Table 4. It is also very apparent that the payload size increases, the PSNR value decreases. But the image quality is still preserved.

Table 4: Embedded payload size vs. PSNR of embedded "Lena" image

Payload Size (bits)	2520	36216	42264	56840	71400	85960
PSNR (dB)	66.7	61.93	54.19	51.56	49.94	48.76

Based on these aspects, it is observed that the proposed method is good in terms of lower computational complexity, less blurring, high data embedding capacity and greater image quality.

5 Conclusion

The proposed data hiding method is a good candidate for providing security over confidential data. Since this method uses neighbor mean interpolation for generating the cover image, so it provides less blurring and degradation in cover image. And also it uses reversible data hiding method, so that original image can be extracted without impairing the image quality. Due to these reasons, hacker can't predict the hidden data. Hence, it is robust method for providing data security across networks. The histogram of original and stego images are compared and it is clear that there is very less degradation between the original and stego image. The performance and quality of the cover image and the stego image are computed and analyzed in terms of PSNR value. With some performance improvement in the computations it can be made useful in real time day to day transaction security related with e-commerce and other online transaction.

Acknowledgments

The authors are grateful to the principal and management of Noorul Islam University and Mepco Schlenk Engineering College for extending all the facilities and constant encouragement for carrying out this research work. The authors gratefully acknowledge the anonymous reviewers for their valuable comments.

References

[1] M. U. Celik, G. Sharma, A. M. Tekalp, and E. Saber, "Lossless generalized LSB data embedding", *IEEE Transactions on Image Processing*, vol. 14, no. 2, pp. 253–266, 2005.

[2] C. C. Chang, C. C. Lin, and Y. H. Chen, "Reversible data embedding scheme using differences between original and predicted pixel values", *IET Information Security*, vol. 2, no. 2, pp. 35–46, 2008.

[3] S. F. Chiou, I. E. Liao, M. S. Hwang, "A capacity-enhanced reversible data hiding scheme based on SMVQ", *Imaging Science Journal*, vol. 59, no. 1, pp. 17–24, 2011.

[4] W. Hong, T. S. Chen., Y. P. Chang, and C. W. Shiu, "A high capacity reversible data hiding scheme using orthogonal projection and prediction error modification", *Signal Processing*, vol. 90, no. 11, pp. 2911–2922, 2010.

[5] L. C. Huang, L. Y. Tseng, and M. S. Hwang, "The study on data hiding in medical images", *International Journal of Network Security*, vol. 14, no. 6, pp. 301–309, 2012.

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

[7] N. F. Johnson, S. Jajodia, "Exploring steganography: seeing the unseen", *IEEE Computing*, vol. 31, no. 2, pp. 26–34, 1998.

[8] X. Li, M. T. Orchard, "New edge-directed interpolation", *IEEE Transactions on Image Processing*, vol. 10, no. 10, pp. 1521–1527, 2001.

[9] Z. Ni, Y. Q. Shi, N. Ansari, and W. Su, "Reversible data hiding", *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 16, no. 3, pp. 354–362, 2006.

[10] F. Peng, L. Xiaolong, and B. Yang, "Adaptive reversible data hiding scheme based on integer transform", *Signal Processing*, vol. 92, no. 1, pp. 54–62, 2012.

[11] G. J. Simmons, "The prisoners' problem and the subliminal channel", in *CRYPTO'83*, pp. 51–67, 1983.

[12] W. L. Tai, C. M. Yeh, and C. C. Chang, "Reversible data hiding based on histogram modification of pixel differences", *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 19, no. 6, pp. 906–910, 2009.

[13] C. C. Thien, J. C. Lin, "A simple and high-hiding capacity method for hiding digit-by-digit data in images based on modulus function", *Pattern Recognition*, vol. 36, pp. 2875–2881, 2003.

[14] J. Tian, "Reversible data embedding using a difference expansion", *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 13, no. 8, pp. 890–896, 2003.

[15] P. Tsai, Y. C. Hu, and H. L. Yeh, "Reversible image hiding scheme using predictive coding and histogram shifting", *Signal Processing*, vol. 89, no. 6, pp.1129–1143, 2009.
[16] R. Z. Wang, C. F. Lin, and J. C. Lin, "Image hiding by optimal LSB substitution and genetic algorithm", *Pattern Recognition*, vol. 34, no. 3, pp. 671–683, 2001.
[17] N. I. Wu and M. S. Hwang, "Data hiding: Current status and key issues", *International Journal of Network Security*, vol. 4, no. 1, pp. 1–9, 2007.

S. Maria Celestin Vigila completed her B.E. in Computer Science and Engineering in 1996 and M.E. in Computer Science and Engineering in 1999. She completed her Ph.D. in the area of data security from Anna University, Chennai. She is currently Associate Professor in the Department of Information Technology, Noorul Islam University, Kumaracoil and member of ISTE and IET. She is the reviewer for quite a few peer reviewed international journals. Her research interest includes cryptography and network security, wireless networks and information hiding.

K. Muneeswaran is Professor and Head in the Department of Computer Science and Engineering, Mepco Schlenk Engineering College, Sivakasi. His area of interest includes image analysis, computer networks, neural networks, security, grid and cloud computing. He contributed to many funded research projects. Also, he is the reviewer for quite a few peer reviewed international journals.

Cryptanalysis of Two Efficient Password-based Authentication Schemes Using Smart Cards

Ying Wang and Xinguang Peng
(Corresponding author: Xinguang Peng)
Department of computer science and technology, Taiyuan University of Technology, Taiyuan 030024, China
(Email: sxgrant@126.com)
(Received May. 31, 2013; revised and accepted Jan. 28 & Mar. 14, 2014)

Abstract

In 2011, Kumar *et al.* proposed an efficient password authentication scheme using smart cards to overcome the security flaws in Liao *et al.* scheme. However, in this paper, we point out that Kumar *et al.*'s scheme actually has various defects been overlooked, such as no provision of forward secrecy, poor repairability and practicality. More recently, Ramasamy and Muniyandi presented an efficient two-factor scheme based on RSA and this scheme is claimed to have a number of merits over existing schemes. Notwithstanding their ambitions, Ramasamy-Muniyandi's scheme is vulnerable to user impersonation attack, and it actually is equivalent to a verifier-table-based scheme, which discourages any use of the scheme for practical applications.

Keywords: Authentication protocol, cryptanalysis, impersonation attack, RSA, smart card

1 Introduction

With the increasing need of accessing remote digital services and protecting electronic transactions, password-based authentication that enable two or more parties sharing memorable passwords to securely communicate over an open channel are gaining popularity due in large part to its practical significance. Its feasibility was investigated as early as the work of Lamport [21], and this initial study has been followed by various proposals, including ones employing multi-application smart cards, [4, 6, 7, 10, 16, 18, 24, 26, 36, 37, 42, 46, 47, 55].

In such schemes, two participants, i.e. a server S and a user U, are involved. In the beginning, U submits her identity ID and password PW to S over a secure channel, and upon receiving the registration request, S issues a smart card to U with the smart card being personalized with some initial security parameters [15, 32]. This phase is called the registration phase and is carried out only once for each client. With the smart card obtained, U can get access to S by employing the login-and-authentication phase. This phase can be carried out as many times as demanded. Besides registration phase and login-and-authentication phase, there may be additional phases, such as the password change phase used when U wants to change her password, and the user eviction phase is used to delete an expired or malicious account.

In 2000, Peyravian and Zunic [34] proposed two user authentication schemes which only employ lightweight hash functions, and thus these two schemes are simple and efficient to be implemented on resource-constrained smart cards. Unfortunately, Peyravian-Zunic's schemes are found vulnerable to various attacks, such as off-fline password guessing attack, stolen-verifier attack and denial-of-service attack, by Hwang and Yeh in 2002 [14]. To overcome the defects in Peyravian-Zunic's schemes, a number of enhanced versions [3, 30] are subsequently put forward. One common feature among these schemes is that, a password-verifier table is stored on the authentication server. As stated by Chen and Lee [5], these schemes in [3, 14, 30, 34] invariably suffer from the risk of modified-verifier-table attack and the cost of protecting and maintaining the verifier table on remote server. If this password-verifier table is stolen by the adversary or leaked by accident, the entire system will be completely broken. Accordingly, intensive research has been made to cope with this problem [12, 18, 22, 28, 48, 50], yet most of the previous schemes are found prone to various issues on both security and performance aspects [13, 23, 25, 27, 31, 32, 40, 41, 45].

As stated by a comprehensive work [44], an important reason for the failure of previous schemes is that, in most of these previous studies, the authors demonstrate attacks on problematic schemes and advance new proposals with claims of the superior aspects of their schemes, and ignore benefits that their schemes fail to provide. Accordingly, a comprehensive and reasonable evaluation metric is of particular importance. In 2006 Liao *et al.* [29] first proposed ten requirements for evaluating a password authentication, and then presented a new scheme using smart cards for password authentication over insecure networks. Liao *et al.* argued that their scheme can satisfy all the ten requirements and thus is immune to

various attacks. Although this scheme possesses many admired features, particularly, no verifier table is needed on the server and a user can freely change her password without interaction with the remote server. However, some security loopholes of this scheme are shortly pointed out by Xiang *et al.* [52].

To remedy the defects identified in Liao *et al.*'s scheme, Kumar *et al.* [20] further put forward an improved scheme in 2011. This scheme is claimed to have enhanced security and could maintain all the advantages of the original scheme and be free from the attacks pointed out by Xiang *et al.* [52]. Notwithstanding their claims, we will report that this scheme still has serval serious defects: (1) it cannot preserve forward secrecy; (2) it has poor repairability; (3) it is not user friendly.

In 2012, Ramasamy and Muniyandi [35] also reported that previous two-factor authentications are far from practicality, and accordingly they put forward an efficient RSA-based password authentication scheme with smart card, which is claimed to be well-suited for practical applications. Their schemes are not only very efficient, but also can withstand various sophisticated attacks such as parallel session attack, denial of service attack and smart card loss attack, and the server has no need to maintain a sensitive password table for authenticating users. However, in this short paper, we will show that Ramasamy-Muniyandi's protocol cannot even attain the basic goal of user authentication by demonstrating its vulnerability to user impersonation attack, in which an adversary does not need any credentials of the legitimate user but just a protocol transcript. Moreover, we reveal that this scheme actually is equal to a password-table-based scheme by presenting a reduction to absurdity.

The rest of this paper is organized as follows: in Section 2, we review Kumar *et al.*'s scheme. Section 3 describes the defects of Kumar *et al.*'s scheme. Then, we turn to review and analyze masamy-Muniyandi's scheme in Section 4 and Section 5, respectively. Finally, the conclusion is drawn in Section 6.

2 Review of Kumar et al.'s Scheme

In this Section, we briefly review the remote user authentication scheme proposed by Kumar *et al.* [20]. Their scheme is composed of four phases: registration, login, authentication, and password change. The notations and descriptions used throughout this paper are summarized in Table 1 and we will follow the notations in Kumar *et al.*'s scheme as closely as possible.

2.1 Initialization Phase

In this phase, AS first selects a large prime number p. Without loss of generality, p is large enough, e.g., at least 1024 bits. Besides, AS selects a secure one-way hash function $h(\cdot)$ and a long secret key x. The details of this phase are described in the following.

Table 1: Notations and abbreviations

Symbol	Description
U_i	i^{th} user
AS	remote authentication server
$\mathcal{M}$	malicious attacker
ID_i	identity of user U_i
PW_i	password of user U_i
x	the secret key of remote server AS
S_{key}	the session key
$h(\cdot)$	collision free one-way hash function
$\oplus$	the bitwise XOR operation
$\|$	the string concatenation operation
$\rightarrow$	a common (insecure) channel
$\Rightarrow$	a secure channel

2.2 Registration Phase

The registration phase involves the following operations:

1) U_i chooses her ID_i and PW_i, generates a random number b and computes $h(b\|PW_i)$.

2) $U_i \Rightarrow AS$: $\{ID_i, h(b\|PW_i)\}$.

3) AS checks the format of ID_i and computes $A_1 = h(ID_i)^{h(b\|PW_i)} \bmod p$, $A_2 = (A_1)^{K(x)} \bmod p$, $EA_2 = A_2 \oplus h(b\|PW_i)$, $B = (h(ID_i))^x \bmod p$, $B_K = K(B)$ and $EB_K = B_K \oplus h(b\|PW_i)$.

4) $AS \Rightarrow U_i : SC$ containing $\{A_1, EA_2, EB_K, p, h(\cdot)\}$.

2.3 Login Phase

When U_i wants to login to AS, the following operations will be performed:

1) U_i inserts her smart card into a card reader and submits her identity ID_i, password PW_i and the random number b^*;

2) SC computes $A_1^* = h(ID_i^*)^{h(b^*\|PW_i^*)} \bmod p$ and checks if $A_1^* \neq A_1$. If the equality does not hold, the login request is rejected by the smart card. Otherwise, SC proceeds to the next step.

3) SC computes $A_2 = EA_2 \oplus h(b\|PW_i)$, $B_K = EB_K \oplus h(b\|PW_i)$, $A_3 = A_2 \oplus h(B_K\|T_{U1}), C_1 = R \oplus h(B_K \|T_{U1})$, $C2 = (A_2, B_K)^R \bmod p$ and $C_3 = h(C_2\|T_{U1})$, where R is a random number.

4) $U_i \rightarrow AS$: Login request $\{ID_i, A_3, C_1, C_3, T_{U1}\}$.

It should be noted that, as with many commercial cards, if U_i fails to enter the correct triple $\{ID_i, PW_i, b\}$ and the number of failed attempts exceeds a predefined value, then SC denies to work further and displays need for re-registration.

2.4 Authentication Phase

After receiving the login request from user U_i, S performs the following operations:

1) S checks the validity of ID_i and that $T_{AS1} - T_{U1} \leq \Delta T$, where T_{AS1} is the time when the login request was received. If either is invalid, the login request is rejected. Otherwise, S performs the following operations.

2) Computes $B_K = K(B) = K[(h(ID_i))^x \bmod p]$, $A_2^* = A_3 \| h(B_K \| T_{U1})$ and $R^* = C_1 \oplus h(B_K \| T_{U1})$.

3) Computes $C_2^* = (A_2^* \| B_K)^{R^*} \bmod p$ and $C_3^* = h(C_2^* \| T_{U1})$. If $C_3^* \neq C_3$ then rejects the login request.

4) Computes $D_1 = S \oplus h(A_2 \| T_{AS2}), D_2 = (C_2)^S \bmod p$ and $D_3 = h(D_2 \| T_{AS2})$, where S is a random number chosen by AS from Z_p^*.

5) $AS \rightarrow U_i : \{D_1, D_3, T_{AS2}\}$. On receiving the response from AS, SC performs as follows:

   a. Checks whether $T_{U2} - T_{AS2} \leq \Delta T$, where T_{U2} is the time when the response was received. If so, then extracts $S^* = D_1 \oplus h(A_2 \| T_{AS2})$.

   b. Computes $D_2^* = (C2)^{S^*} \bmod p$ and $D_3^* = h(D_2^* \| \ T_{AS2})$. If $D_3^* = D_3$, then the legality of AS is confirmed.

6) After authenticating each other, U_i and AS use the same session key $S_{key} = h(D_2 \| A_2 \ \| BK \ \| R \ \| S \ \| T_{U1} \ \| T_{AS2})$ for further communications.

2.5 Password Change Activity

When U_i wants to change the old password PW_i to a new one, this phase will be involved and U_i does not need to interact with AS.

1) U inserts her SC into the smart card device and then keys her identity ID_i^*, password PW_i^*, and random number b^*; and requests SC to change the password.

2) Computes $A_1^* = h(ID_i^*)^{h(b^* \| PW_i^*)} \bmod p$. If $A_1^* = A_1$, then U is allowed to enter the new password PW_i^{**};

3) Extracts $A_2 = EA_2 \oplus h(b^* \| PW_i^*)$, $B_K = EB_K \oplus h(b^* \| PW_i^*)$ and $A_1^{**} = h(ID^*)^{h(b^* \| PW_i^{**})}$;

4) Computes $A_2^{**} = A_2^{(h^{-1}(b^* \| PW_i^*))(h(b^* \| PW_i^{**}))} \bmod p$, $EA_2^{**} = A_2^{**} \oplus h(b^* \| PW_i^{**})$ and $EB_K^{**} = B_K \oplus h(b^* \| PW_i^{**})$;

5) Replaces A_1, EA_2 and EB_K with A_1^{**}, EA_2^{**}, and BK^{**} respectively.

3 Cryptanalysis of Kumar et al.'s Scheme

In this Section we will show that Kumar et al.'s scheme [20] fails to provide forward secrecy, has poor repairability and is not user-friendly, which make this scheme unpractical. There are three assumptions of the adversary's capabilities clearly made in Kumar et al.'s scheme, and we summarize them as follows:

Assumption 1. *The malicious attacker $\mathcal{M}$ can eavesdrop, insert, delete, alter, intercept or block any messages transmitted in the channel. In other words, $\mathcal{M}$ has total control over the communication channel between the user U and the remote server S, this is consistent with the Dolev-Yao standard distributed computing adversary model [9];*

Assumption 2. *The malicious attacker $\mathcal{M}$ is able to extract the secret security parameters stored in the smart card when the user's smart card is in $\mathcal{M}$'s possession. This assumption is reasonable according to the recent research results on side-channel attack techniques [1, 2, 17, 33].*

Assumption 3. *The malicious attacker $\mathcal{M}$ can offline enumerate the password space. For user-friendliness, most schemes (e.g., the schemes in [11, 23, 27, 31]) facilitate the users to select their own password at will during the password change phase and registration phase and the users often choose passwords which are easily remembered for their convenience, and these easily-remembered passwords are weak and fall into a small dictionary [8, 51].*

It is worth noting that the above three assumptions are also explicitly made in most of the latest works [13, 27, 32, 38, 39, 40, 41, 45], and indeed reasonable as justified in [46, 54]. Based on the above assumptions, in the following discussions of the security flaws of Kumar et al.'s scheme, we assume that an attacker can extract the secret values $\{A_1, EA_2, EB_k, p\}$ stored in the legitimate user's smart card, and the attacker can also intercept or block the login request $\{ID_i, A_3, C_1, C_3, T_{U1}\}$ sent out by U_i and the reply message $\{D_1, D_3, T_{AS2}\}$ sent out by the server AS.

3.1 Failure to Achieve Forward Secrecy

As noted in [43, 53], forward secrecy is an important property of remote user authentication schemes for limiting the effects of eventual failure of the entire system in case the long-term private key(s) of the authentication server is compromised (leaked or stolen). A scheme with perfect forward secrecy assures that, even if the server's long-term key is compromised, the previously established session keys will not be compromised.

When analyzing their scheme, Kumar *et al.* argued that "if the secret key x of AS is revealed accidentally,

even in possession of U_i's smart card, $\mathcal{M}$ can neither behave like legal AS nor like a legal U_i", and hence this scheme is claimed to provide forward secrecy. Firstly, we have to say that Kumar et al. have misunderstood the meaning of forward secrecy. Actually, as stated in [19, 43], forward secrecy has nothing to do with impersonation but relates to session keys. With this notion misunderstood, their scheme, of course, cannot achieve this important property.

Supposing an attacker $\mathcal{M}$ has obtained the master secret key x from the compromised server and eavesdropped the transcripts $\{ID_i, A_3, C_1, C_3, T_{U1}, D_1, D_3, T_{AS2}\}$ during U_i and AS's jth authentication process from the open channel. $\mathcal{M}$ can compute the session key of U_i and AS's jth encrypted communication as follows:

Step 1. Computes $B_K = K(B) = K[(h(ID_i))^x \bmod p]$, where ID_i is previously obtained by eavesdropping on the public channel.

Step 2. Computes $A_2 = A_3 \| h(B_K \| T_{U1})$, $R = C_1 \oplus h(B_K \| T_{U1})$, where A_3 and T_{U1} is previously obtained by eavesdropping on the public channel;

Step 3. Computes $C_2 = (A_2 \| B_K)^R \bmod p$;

Step 4. Computes $S = D_1 \oplus h(A_2 \| T_{AS2})$, where T_{AS2} is previously obtained by eavesdropping on the public channel;

Step 5. Computes $D_2 = (C_2)^S \bmod p$;

Step 6. Computes the jth session key $S^j_{key} = h(D_2 \| A_2 \| BK \| R \| S \| T_{U1} \| T_{AS2})$.

Once the session key SK^j is obtained, the whole jth session will be completely exposed to $\mathcal{M}$. Therefore, as opposed to Kumar et al.'s claim, forward secrecy is not provided in their scheme.

3.2 Poor Practicality

In Kumar *et al.*'s scheme, the user has to input three items, i.e. ID_i, PW_i and b when login. As stated in [20], b is a random number generated by U_i when registration. If it is large (and really random), it will be very hard for the user to remember and it is most likely that U_i may forget this long and random number if she does not frequently use the system, which will render the scheme completely unusable. However, if it is not large enough (i.e. not of high entropy and drawn from a small dictionary $\mathcal{D}_b$), it can be easily guessed as with guessing the password, and this scheme will be vulnerable to offline password guessing attack. In case an attacker $\mathcal{M}$ gets access to U_i's smart card for a period of time, according Assumption 2, $\mathcal{M}$ can extract the secret values $\{A_1, EA_2, EB_k, p\}$ stored in the legitimate user's smart card. Then, an offline password guessing attack can be launched as follows:

Step 1. Guesses the value of PW_i to be PW_i^* from a dictionary space $\mathcal{D}_{pw}$, the value of b to be b_i^* from a dictionary space $\mathcal{D}_b$;

Step 2. Computes $A_1^* = h(ID_i)^{b^* \| PW_i^*}$;

Step 3. Verifies the correctness of PW_i^* and b^* by checking if the computed A_1^* is equal to the revealed A_1, where A_1 is extracted from U_i's smart card;

Step 4. Repeats the above steps until the correct value of PW_i is found.

Let $|\mathcal{D}_{pw}|$ denote the number of passwords in the password space $\mathcal{D}_{pw}$, $|\mathcal{D}_b|$ denote the number of items in $\mathcal{D}_b$. The running time of the above attack procedure is $\mathcal{O}(|\mathcal{D}_{pw}| * |\mathcal{D}_{pw}| * T_H)$, where T_H is the running time for hash operation. As $|\mathcal{D}_{pw}|$ and $|\mathcal{D}_b|$ are very limited in practice [8, 51], the above attack can be completed in polynomial time.

3.3 Poor Repairability

In Kumar *et al.*'s scheme, when a user suspects (or realizes) that she has been impersonated by an attacker, however, even if U_i changes her password to a new one, such a fraud can not be prohibited. Since A_1 is uniquely determined by U_i's identity ID_i and AS's permanent secret key x, AS can not change A_1 for U_i unless either ID_i or x is changed. Unfortunately, since ID_i is tied to U_i uniquely in most application systems and it is not reasonable to change ID_i. Furthermore, it is also impractical and inefficient to change x to recover the security for U_i, since x is commonly used for all users rather than specifically used for only one user.

4 A Brief Review of Ramasamy-Muniyandi's Scheme

In this Section, we briefly review the remote user authentication scheme proposed by Ramasamy and Muniyandi [35] in 2012. Their scheme is based on RSA and involves three parties, i.e. the user U_i, the server S and the key information center (KIC). KIC is responsible for registration only and does not participate in the authentication process. Their scheme consists of three phases: the registration phase, the login phase and the authentication phase. In the following, we employ the notations listed in Table 1 and follow the original notations in [35] as closely as possible.

4.1 Registration Phase

User U_i chooses her identity ID_i and password PW_i, and submits them to KIC. For issuing a smart card to user U_i, KIC performs the registration steps:

1) Generates an RSA key pair, namely a private key d and a public key (e, n), $ed = 1 \bmod \psi(n)$, $n = pq$, where p and q are two large primes of nearly the same length. KIC publishes (e, n) and keeps d secret.

2) Determines an integer g, which g is a primitive in both GF_p and GF_q.

3) Generates the smart card identifier CID_i of U_i and calculates security parameter $W_i = ID_i^{CID_i \times d} \bmod n$.

4) Computes $V_i = g^{PW_i \times d \times T_R} \bmod n$, where T_R is the user's registration time. This value is unique for every user and maintained by the server AS. In other words, AS keeps an entry $\{ID_i, T_R\}$ for each registered user U_i.

5) $AS \Rightarrow U_i$: A smart card containing security parameters $\{n, e, CID_i, W_i, V_i, h(\cdot)\}$.

4.2 Login Phase

When U_i wants to login to S, she inserts her smart card into a card reader and keys ID_i and PW_i. Then the smart card will perform the following steps:

1) Generates a random number r and calculate $X_i = g^{PW_i \times r} \bmod n$ and $Y_i = W_i \times V_i^{r \times T} \bmod n$.

2) $U_i \rightarrow S$: $\{(ID_i, CID_i, X_i, Y_i, n, e, g, T_u)\}$.

4.3 Authentication Phase

On receiving the login request, the server S performs the following steps:

1) Checks whether ID_i is a valid user identity and CID_i is a legal smart card identity. If either is not valid, AS rejects the login request.

2) Checks whether $T_s - T_u \leq \Delta T$, where T_s is the time when the login request is received and ΔT is the legal time interval due to transmission delay, if not, then AS rejects the login request.

3) Evaluates the equation $Y_i^e = ID_i^{CID_i} \times X_i^{T_u \times T_R} \bmod n$, where T_u is the login request time and T_R is the registration time of U_i.

4) If any one of the above results is negative, then login request is rejected. Otherwise, the login request is accepted.

5) If the login request is rejected three times then the user account will be automatically locked and she has to contact the server to unlock the account.

5 Cryptanalysis of Ramasamy-Muniyandi's Scheme

In this Section, we will discuss the flaws of Ramasamy-Muniyandi's scheme. Note that the three assumptions listed in Section 3 are also clearly made in [35]. This scheme is simple and elegant, however, after careful examination, we find it cannot achieve the basic goal of user authentication. Besides, their scheme has an inherent design flaw in the registration phase and it actually is equal to a verifier-table-based scheme. The identified defects discourage any use of the scheme for practical applications.

5.1 User Impersonation Attack

In the following, we will show how an attacker $\mathcal{M}$ without any credentials (i.e., the password and the smart card) of U_i can successfully impersonate U_i to login to SA and freely enjoy the services.

Step 1. Intercepts and block a login request $\{(ID_i, CID_i, X_i, Y_i, n, e, g, T_u)\}$ of the user U_i from the public communication channel;

Step 2. Computes $T_u' = \varepsilon T_u$, where ε is a small real number chosen by $\mathcal{M}$ in such a way that T_u' is a valid timestamp in the near future;

Step 3. $\mathcal{M} \rightarrow AS$: $\{(ID_i, CID_i, X_i^{\varepsilon}, Y_i, n, e, g, T_u')\}$.

Step 4. The server AS checks the validity of the timestamp T_u' by checking $T_s - T_u' \leq \Delta T$, where T_s denotes the server's current timestamp. Then the server AS checks $Y_i^e \stackrel{?}{=} ID_i^{CID_i} \times (X_i^{\varepsilon})^{T_u \times T_R} \bmod n$.

Now we show that in Step 4, AS will find no abnormality, because

$$\begin{aligned} Y_i^e &= (W_i \times V_i^{r \times T_u'}) \bmod n \\ &= ID_i^{CID_i} \times g^{PW_i \times r \times T_R \times T_u'} \bmod n \\ &= ID_i^{CID_i} \times g^{PW_i \times r \times T_R \times \varepsilon \times T_u} \bmod n \\ &= ID_i^{CID_i} \times g^{(PW_i \times r)^{T_R \times \varepsilon \times T_u}} \bmod n \\ &= ID_i^{CID_i} \times (X_i^{\varepsilon})^{T_R \times T_u} \bmod n. \end{aligned}$$

On successful verification, the server AS accepts the forged login authentication request. Therefore, the attacker $\mathcal{M}$ can impersonate as the legitimate user without any cryptographic credentials, which breaches the soundness of the underlying authentication scheme.

5.2 The Problem of Storing Parameter T_R

In this Section, we demonstrate another serious defect in Ramasamy-Muniyandi's scheme. In the registration phase, AS keeps an entry $\{ID_i, T_R\}$ for each registered user U_i. At first glance, T_R is not the user's password and the store of such an entry does not violate the basic goal of no password-verifier table. However, T_R actually is as critical as the password, and Ramasamy-Muniyandi's scheme equals to a scheme with password-verifier table. We prove this by contradiction.

If Ramasamy-Muniyandi's scheme is a scheme with no "password-verifier table", then the disclosure of T_R alone (i.e., U_i's smart card and password, server's private key x are still secure) will pose no threat to the security of the scheme. Now we assume U_i's entry on the server has disclosed and been obtained by the attacker $\mathcal{M}$.

If $\texttt{gcd}\ (T_R, e) = 1$, $\mathcal{M}$ can impersonate as U_i by performing the following steps:

Step 1. Intercepts and blocks a login request $\{ID_i, CID_i, X_i, Y_i, n, e, g, T_u\}$ of the user U_i from the public communication channel.

Step 2. Reads the current timestamp T_u and checks if $\texttt{gcd}\ (T_R \times T_u, e) = 1$. If it holds, proceeds to the next step. Otherwise, $\mathcal{M}$ repeats this step.

Step 3. Runs the Extended Euclidean algorithm to compute two integers a and b such that $a \times e + b \times T_u \times T_R = 1 (\text{in } \mathbb{Z})$.

Step 4. Computes $X_i' = (ID_i^{CID_i})^{-b} \ \texttt{mod}\ n$ and $Y_i' = (ID_i^{CID_i})^{a} \ \texttt{mod}\ n$.

Step 5. $\mathcal{M} \rightarrow AS$: $\{(ID_i, CID_i, X_i', Y_i', n, e, g, T_u)\}$.

Step 6. The server AS checks the validity of the timestamp T_u by checking $T_s - T_u \leq \Delta T$, where T_s denotes the server's current timestamp. Then the server AS checks $(Y_i')^e \stackrel{?}{=} ID_i^{CID_i} \times (X_i')^{T_u \times T_R} \ \texttt{mod}\ n$.

We give a few remarks on the above attack. Firstly, in Step 3, $\mathcal{M}$ can definitely find a and b, for the value of T_u is chosen in such a way that $\texttt{gcd}\ (T_R \times T_u, e) = 1$. Secondly, in Step 6, the server AS will accept, which is justified by the following equalities:

$$\begin{aligned}(Y_i')^e &= (ID_i^{CID_i})^{ae} \ \texttt{mod}\ n \\ &= (ID_i^{CID_i})^{(-b)\times T_u \times T_R} \ \texttt{mod}\ n \\ &= ID_i^{CID_i} \times (ID_i^{CID_i})^{-b \times T_u \times T_R} \ \texttt{mod}\ n \\ &= ID_i^{CID_i} \times (ID_i^{CID_i \times (-b)})^{T_u \times T_R} \ \texttt{mod}\ n \\ &= ID_i^{CID_i} \times X_i^{T_u \times T_R} \ \texttt{mod}\ n.\end{aligned}$$

The above attack procedure has shown that if $\texttt{gcd}\ (T_R, e) = 1$, $\mathcal{M}$ can impersonate as U_i with the help of the leaked T_R. We now show that, the above attack has a success rate about 60% due to the following two facts: (1) The probability of $\texttt{gcd}\ (T_R, e) = 1$ is about $6/\pi^2 \approx 0.6$ [49]; (2) T_R and e are chosen by different parties, and thus they are independent.

The above analysis demonstrates that $\mathcal{M}$ can impersonate as U_i with remarkably high probability (i.e., a success rate about 60%) in case T_R is leaked. Consequently, the leakage of the $\{ID_i, T_R\}$ table does endanger the security of the scheme and it should be well kept secret, which invalidates the claim of a "no verifier table" scheme. As stated in the introduction, it is greatly undesirable for the server to maintain and protect a verifier table.

6 Conclusion

Two-factor authentication is an important mechanism for remote login systems that enables the server and its users to authenticate each other. In this paper, we first pointed out that Kumar *et al.*'s scheme is really impractical by demonstrating three serious defects. Then, we illustrated that Ramasamy-Muniyandi's RSA-based authentication scheme is prone to a user impersonation attack and equal to a verifier-based scheme. In our security analysis, we employed the number theory that two random (or independently chosen) numbers are relatively prime with a probability about $6/\pi^2 \approx 0.6$. As for future work, we are considering to design two-factor authentication schemes with formal security.

Acknowledgments

The authors would like to thank the anonymous reviewers for their valuable comments and constructive suggestions. This research was in part supported by the Natural Science Foundation for Young Scientists of Shanxi Province under Grant No. 2012021011-3, National Natural Science Foundation of Shanxi Province under Grant No. 2009011022-2 and Shanxi Scholarship Council of China under Grant No. 2009-28.

References

[1] F. Amiel, B. Feix, and K. Villegas, "Power analysis for secret recovering and reverse engineering of public key algorithms," in *Proceedings of SAC'07*, LNCS 4876, pp. 110–125, Springer, 2007.

[2] J. Balasch, B. Gierlichs, R. Verdult, L. Batina, and I. Verbauwhede, "Power analysis of atmel crypto Memory–Recovering keys from secure EEPROMs," in *Topics in Cryptology (CT-RSA'12)*, pp. 19–34, Springer, 2012.

[3] Y. F. Chang, C. C. Chang, and L. I. U. Yi-Long, "Password authentication without the server public key," *IEICE Transactions on Communications*, vol. 87, no. 10, pp. 3088–3091, 2004.

[4] T. H. Chen, H. C. Hsiang, and W. K. Shih, "Security enhancement on an improvement on two remote user authentication schemes using smart cards," *Future Generation Computer Systems*, vol. 27, no. 4, pp. 377–380, 2011.

[5] T. H. Chen and W. B. Lee, "A new method for using hash functions to solve remote user authentication," *Computers & Electrical Engineering*, vol. 34, no. 1, pp. 53–62, 2008.

[6] H. R. Chung, W. C. Ku, and M. J. Tsaur, "Weaknesses and improvement of wang et al.'s remote user password authentication scheme for resource-limited environments," *Computer Standards & Interfaces*, vol. 31, no. 4, pp. 863–868, 2009.

[7] M. L. Das, A. Saxena, and V. P. Gulati, "A dynamic ID-based remote user authentication scheme," *IEEE Transactions on Consumer Electronics*, vol. 50, no. 2, pp. 629–631, 2004.

[8] M. Dell'Amico, P. Michiardi, and Y. Roudier, "Password strength: An empirical analysis," in *Proceedings of Infocom'10*, pp. 1–9, Mar. 2010.

[9] D. Dolev and A. Yao, "On the security of public key protocols," *IEEE Transactions on Information Theory*, vol. 29, no. 2, pp. 198–208, 1983.

[10] D. He, J. Chen, and J. Hu, "An ID-based client authentication with key agreement protocol for mobile client-server environment on ECC with provable security," *Information Fusion*, vol. 13, no. 3, pp. 223–230, 2012.

[11] D. He, J. Chen, and J. Hu, "Improvement on a smart card based password authentication scheme," *Journal of Internet Technology*, vol. 13, no. 3, pp. 38–42, 2012.

[12] D. He, J. Chen, and R. Zhang, "Weaknesses of a dynamic ID-based remote user authentication scheme," *International Journal of Electronic Security and Digital Forensics*, vol. 3, no. 4, pp. 355–362, 2010.

[13] D. He and S. Wu, "Security flaws in a smart card based authentication scheme for multi-server environment," *Wireless Personal Communications*, vol. 70, no. 1, pp. 323–329, 2013.

[14] J. J. Hwang and Y. E. H. Tzu-Chang, "Improvement on Peyravian-Zunic's password authentication schemes," *IEICE Transactions on Communications*, vol. 85, no. 4, pp. 823–825, 2002.

[15] M. S. Hwang, S. K. Chong, and T. Y. Chen, "DoS-resistant ID-based password authentication scheme using smart cards," *Journal of Systems and Software*, vol. 83, no. 1, pp. 163–172, 2010.

[16] M. S. Hwang and L. H. Li, "A new remote user authentication scheme using smart cards," *IEEE Transactions on Consumer Electronics*, vol. 46, no. 1, pp. 28–30, 2000.

[17] T. Kasper, D. Oswald, and C. Paar, "Side-channel analysis of cryptographic RFIDs with analog demodulation," in *Proceedings of RFIDSec'12*, LNCS 7055, pp. 61–77, Springer, 2012.

[18] M. K. Khan, S. K. Kim, and K. Alghathbar, "Cryptanalysis and security enhancement of a more efficient & secure dynamic ID-based remote user authentication scheme'," *Computer Communications*, vol. 34, no. 3, pp. 305–309, 2011.

[19] H. Krawczyk, "HMQV: A High-Performance secure Diffie-Hellman protocol," in *Advances in Cryptology (Crypto'05)*, LNCS 3621, pp. 546–566, 2005.

[20] M. Kumar, M. K. Gupta, and S. Kumari, "An improved efficient remote password authentication scheme with smart card over insecure networks," *International Journal of Network Security*, vol. 13, no. 3, pp. 167–177, 2011.

[21] L. Lamport, "Password authentication with insecure communication," *Communications of the ACM*, vol. 24, no. 11, pp. 770–772, 1981.

[22] C. C. Lee, M. S. Hwang, and I. E. Liao, "Security enhancement on a new authentication scheme with anonymity for wireless environments," *IEEE Transactions on Industrial Electronics*, vol. 53, no. 5, pp. 1683–1687, 2006.

[23] C. C. Lee, C. T. Li, and R. X. Chang, "A simple and efficient authentication scheme for mobile satellite communication systems," *International Journal of Satellite Communications Networking*, vol. 30, no. 1, pp. 29–38, 2012.

[24] C. C. Lee, T. H. Lin, and R. X. Chang, "A secure dynamic ID based remote user authentication scheme for multi-server environment using smart cards," *Expert Systems with Applications*, vol. 38, no. 11, pp. 13863–13870, 2011.

[25] C. C. Lee, C. H. Liu, and M. S. Hwang, "Guessing attacks on strong-password authentication protocol," *International Journal of Network Security*, vol. 15, no. 1, pp. 64–67, 2013.

[26] C. T. Li and C. C. Lee, "A robust remote user authentication scheme using smart card," *Information Technology and Control*, vol. 40, no. 3, pp. 236–245, 2011.

[27] C. T. Li and C. C. Lee, "A novel user authentication and privacy preserving scheme with smart cards for wireless communications," *Mathematical and Computer Modelling*, vol. 55, no. 1, pp. 35–44, 2012.

[28] C. T. Li, C. C. Lee, C. J. Liu, and C. W. Lee, "A robust remote user authentication scheme against smart card security breach," in *Proceedings of 25th Annual IFIP Conference on Data and Applications Security and Privacy (DBSec '11)*, LNCS 6818, pp. 231–238, 2011.

[29] I. E. Liao, C. C. Lee, and M. S. Hwang, "A password authentication scheme over insecure networks," *Journal of Computer and System Sciences*, vol. 72, no. 4, pp. 727–740, 2006.

[30] C. L. Lin and T. Hwang, "A password authentication scheme with secure password updating," *Computers & Security*, vol. 22, no. 1, pp. 68–72, 2003.

[31] C. G. Ma, D. Wang, and Q. M. Zhang, "Cryptanalysis and improvement of sood et al.s dynamic ID-Based authentication scheme," in *Proceedings of International Conference on Distributed Computing and Internet Technology (ICDCIT'12)*, LNCS 7154, pp. 141–152, 2012.

[32] C. G. Ma, D. Wang, and S. D. Zhao, "Security flaws in two improved remote user authentication schemes using smart cards," *International Journal of Communication Systems*, vol. 27, no. 10, pp. 2215–2227, 2014.

[33] T. S. Messerges, E. A. Dabbish, and R. H. Sloan, "Examining smart-card security under the threat of power analysis attacks," *IEEE Transactions on Computers*, vol. 51, no. 5, pp. 541–552, 2002.

[34] M. Peyravian and N. Zunic, "Methods for protecting password transmission," *Computers & Security*, vol. 19, no. 5, pp. 466–469, 2000.

[35] R. Ramasamy and A. P. Muniyandi, "An efficient password authentication scheme for smart card," *International Journal of Network Security*, vol. 14, no. 3, pp. 180–186, 2012.

[36] J. J. Shen, C. W. Lin, and M. S. Hwang, "A modified remote user authentication scheme using smart cards," *IEEE Transactions on Consumer Electronics*, vol. 49, no. 2, pp. 414–416, 2003.

[37] J. J. Shen, C. W. Lin, and M. S. Hwang, "Security enhancement for the timestamp-based password authentication scheme using smart cards," *Computers & Security*, vol. 22, no. 7, pp. 591–595, 2003.

[38] K. A. Shim, "Security flaws in three Password-Based remote user authentication schemes with smart cards," *Cryptologia*, vol. 36, no. 1, pp. 62–69, 2012.

[39] R. Song, "Advanced smart card based password authentication protocol," *Computer Standards & Interfaces*, vol. 32, no. 5, pp. 321–325, 2010.

[40] S. K. Sood, "An improved and secure smart card based dynamic identity authentication protocol," *International Journal of Network Security*, vol. 14, no. 1, pp. 39–46, 2012.

[41] H. B. Tang, X. S. Liu, and L. Jiang, "A robust and efficient timestamp-based remote user authentication scheme with smart card lost attack resistance," *International Journal of Network Security*, vol. 15, no. 6, pp. 360–368, 2013.

[42] X. Tian, R. W. Zhu, and D. S. Wong, "Improved efficient remote user authentication schemes," *International Journal of Network Security*, vol. 4, no. 2, pp. 149–154, 2007.

[43] D. Wang and C. G. Ma, "Cryptanalysis and security enhancement of a remote user authentication scheme using smart cards," *The Journal of China Universities of Posts and Telecommunications*, vol. 19, no. 5, pp. 104–114, 2012.

[44] D. Wang and C. G. Ma, "Robust smart card based password authentication scheme against smart card loss problem," Cryptology ePrint Archive, Report 2012/439, 2012. (`http://eprint.iacr.org/2012/439.pdf`)

[45] D. Wang, C. G. Ma, and P. Wu, "Secure Password-Based remote user authentication scheme with Non-tamper resistant smart cards," in *Data and Applications Security and Privacy*, LNCS 7371, pp. 114–121, 2012.

[46] D. Wang and P. Wang, "Offline dictionary attack on password authentication schemes using smart cards," in *Proceedings of the 16th Information Security Conference (ISC'13)*, pp. 1–16, 2013.

[47] X. M. Wang, W. F. Zhang, J. S. Zhang, and M. K. Khan, "Cryptanalysis and improvement on two efficient remote user authentication scheme using smart cards," *Computer Standards & Interfaces*, vol. 29, no. 5, pp. 507–512, 2007.

[48] Y. Wang, J. Liu, F. Xiao, and J. Dan, "A more efficient and secure dynamic ID-based remote user authentication scheme," *Computer Communications*, vol. 32, no. 4, pp. 583–585, 2009.

[49] E. Weisstein, "Relatively prime," 2013. (`http://mathworld.wolfram.com/RelativelyPrime.html`)

[50] F. Wen and X. Li, "An improved dynamic ID-based remote user authentication with key agreement scheme," *Computers & Electrical Engineering*, vol. 38, no. 2, pp. 381–387, 2012.

[51] T. Wu, "A real-world analysis of kerberos password security," in *Proceedings of the 1999 ISOC Network and Distributed System Security Symposium*, pp. 1–14, 1999.

[52] T. Xiang, K. Wong, and X. Liao, "Cryptanalysis of a password authentication scheme over insecure networks," *Journal of Computer and System Sciences*, vol. 74, no. 5, pp. 657–661, 2008.

[53] L. Xiong, N. Jianwei, K. Muhammad Khurram, and L. Junguo, "An enhanced smart card based remote user password authentication scheme," *Journal of Network and Computer Applications*, vol. 36, no. 5, pp. 1365–1371, 2013.

[54] J. Xu, W. T. Zhu, and D. G. Feng, "An improved smart card based password authentication scheme with provable security," *Computer Standards & Interfaces*, vol. 31, no. 4, pp. 723–728, 2009.

[55] K. H. Yeh, C. Su, N. W. Lo, Y. Li, and Y. X. Hung, "Two robust remote user authentication protocols using smart cards," *Journal of Systems and Software*, vol. 83, no. 12, pp. 2556–2565, 2010.

Ying Wang received her MS degree in the department of computer science and technology in 2006 from Taiyuan University of Technology, China. She is currently a Ph.D. candidate and lecturer in the department of computer science and technology of Taiyuan University of Technology, China. Her research interests include computer network and security, trusted computing and cryptography.

Xin-Guang Peng received his Ph.D. in computer application technology from the Beijing Institute of Technology, China in 2004. He is a professor in the department of computer science and technology of Taiyuan University of Technology, China. His research interests include computer network and security, trusted computing.

Provable Secure Multi-Proxy Signature Scheme without Bilinear Maps

Namita Tiwari and Sahadeo Padhye
(Corresponding author: Sahadeo Padhye)

Department of Mathematics, Motilal Nehru National Institute Of Technology
Allahabad (U.P.), India.
(Email : sahadeomathrsu@gmail.com)

(Received July 06, 2013; revised and accepted Oct. 10 & Nov. 7, 2013)

Abstract

Multi-proxy signature (MPS) allows an original signer to authorize a group of proxy signers as his proxy agent to sign co-operatively a message. As per the literature, the relative computation cost of a pairing is approximately more than ten times of the scalar multiplication over elliptic curve group which indicates that pairing is a very expensive operation. Moreover the map-to-point function is also very expensive. Therefore, we propose a new MPS scheme without pairings having general cryptographic hash function after formalizing a security model. Our proposal is provable secure and much efficient than previously proposed schemes in practice.

Keywords: Bilinear pairings, digital signature, elliptic curve discrete log problem, multi-proxy signature, random oracle model

1 Introduction

The concept of proxy signature was firstly introduced by Mambo et al. [13], to sign the messages on behalf of original signer. In a proxy signature scheme, an authorized person, called the proxy signer, is delegated by the original signer to generate a proxy signature on behalf of the original signer. To delegate the signing rights, a warrant message is used which consist of the identity of original as well as proxy signer's group, delegation period, information about the message etc. Original signer generates the delegation by signing the message warrant. Proxy signatures can be verified using a modified verification equation such that the verifier can be convinced that the signature is generated by the authorized proxy entity of the original signer. On the other hand, proxy signature is needed in some other forms also that are described in the article [17] in detail. For example, two or more vice presidents can cooperatively make a significant decision or sign an important document on behalf of the president in his absence. MPS is the solution of such a problem which allows the original signer to delegate his/her signing power to a group of proxy signers such that all proxy signers must cooperatively generate a valid proxy signature.

On the other side, if a group of original signer want to authorize a proxy signer to generate a signature on behalf of the original signer group, to handle such a situation in 2000, Yi et al. [26] firstly proposed proxy multi-signature (PMS) scheme. After that, some other variants multi-proxy multi-signature (MPMS) schemes have also been proposed [7].

Since proxy signature appeared, many new proxy signature schemes [4, 5, 6, 8, 19, 21, 27] have been proposed. Motivated by the recent work [6], authors proposed the ID-based proxy multi-signature [16] and multi-proxy multi-signature [18] schemes without pairings. In this paper, we focus on MPS scheme. Till now, many MPS schemes [2, 10, 11, 20, 22, 23, 24, 25] *etc* from bilinear pairings and ElGamal type have been proposed. There are some literatures [1, 6] etc showing that the relative computation cost of a pairing operation is approximately more than ten times of the scalar multiplication over elliptic curve group. In addition, the map-to-point hash function is also very expensive cryptographic operation. Due to bilinear pairings and map-to-point hash function, the above schemes are less efficient and so not very applicable in practice. Therefore, schemes without bilinear pairings in general hash function setting with elliptic curve cryptography would be more appealing in terms of efficiency while maintaining the security.

Elliptic curve cryptography (ECC) was introduced by Koblitz [9] and Miller [14] independently in 1985 using the group of points on an elliptic curve defined over a finite field. Security of the cryptosystem based on ECC relies on elliptic curve discrete log problem (ECDLP). The main advantage of ECC is that it provides the same security level with smaller key size [12] than RSA and ElGamal cryptosystems. Smaller key means less management time

and smaller storage, which supplies convenience to realization by software and hardware. To achieve $1024-bits$ RSA level security, $512-bits$ supersingular elliptic curve and $160-bits$ non-supersingular elliptic curves are used in applications. In general, pairing is defined on the supersingular elliptic curve while the ECC without pairings uses non supersingular elliptic curves.

In this paper, we propose an efficient MPS scheme without bilinear pairings which has smaller key size than pairing based schemes. Proposed scheme is proven secure against adaptive chosen message attack [3] under ECDLP assumption. With the pairing-free realization, proposed scheme is much efficient than previous related schemes from pairings in practice. In addition, it is obviously much efficient in practice than ElGamal based MPS schemes since ECC provides the same security than ElGamal based cryptosystems at less bit parameters.

The rest of this paper is organized as follows. Some preliminary works are given in Section 2. The formal models of MPS scheme is described in Section 3. Our provable secure MPS scheme is presented in Section 4. We analyze the security of proposed scheme in Section 5. Section 6 presents the comparative analysis. Finally, conclusions are given in Section 7.

2 Preliminaries

2.1 Background of Elliptic Curve Group

An elliptic curve E over a prime finite field F_p (denoted by E/F_p) is the set of points (x, y) with $x, y \in F_p$ which satisfy the equation $y^2 = (x^3 + ax + b) \bmod p$, $a, b \in F_p$,point say $-R$. Then $P + Q$ is the reflected point $-R$. There is a together with an extra point $\{\infty\}$ (called the point at infinity). If the discriminant $\Delta = (4a^3 + 27b^2) \bmod p \neq 0$, equivalently, the polynomial $x^3 + ax + b$ has distinct factors then E/F_p is nonsingular i.e it does not have any cusp or node singularity. Therefore, we can define a binary operation (the point addition " + ") on the points of E/F_p as follows: Let $P, Q \in E/F_p$, l be the line joining P and Q (tangent line to E/F_p if $P = Q$), and R, the third point of intersection of l with E/F_p. Let l' be the vertical line through R which intersects the elliptic curve E/F_p at another problem that vertical line through P and $-P$ does not intersect elliptic curve E/F_p at a third point and we need a third point to define $P+(-P)$. Since there is no point in the plane that works, we create an extra point ∞ at infinity. Here ∞ is a point on every vertical line.

Thus elliptic curve with this binary operation " + " forms an additive abelian group $(E/F_p, \text{" + "}) = \{(x, y) : x, y \in F_p, E(x, y) = 0\} \cup \{\infty\}$. Let G be a cyclic additive subgroup of $(E/F_p, \text{"+"})$ with generator P of prime order n.

2.2 Mathematical Formulas for Addition on E/F_p

Suppose that we want to add the points $P_1 = (x_1, y_1)$ and $P_2 = (x_2, y_2)$ on the elliptic curve E as defined above.

Let the line connecting P_1 to P_2 be $L : y = mx + c$. Explicitly, the slope and y-intercept of L are given by

$$\begin{aligned} m &= \begin{cases} \frac{y_2-y_1}{x_2-x_1} \bmod p, & \text{if } P_1 \neq P_2 \\ \frac{3{x_1}^2+a}{2y_1} \bmod p, & \text{if } P_1 = P_2 \end{cases} \\ c &= (y_1 - mx_1) \bmod p. \end{aligned}$$

Now we find the intersection of E/F_P: $y^2 = (x^3+ax+b)$ $a, b \in F_p$, and L: $y = mx + c$ by solving $(mx + c)^2 = x^3 + ax + b$ under modulo p. We already know that x_1 and x_2 are solutions, so we can find the third solution x_3 by comparing the two sides of $x^3 + ax + b - (mx + c)^2 = (x-x_1)(x-x_2)(x-x_3) \bmod p$. Equating the coefficients of x^2, gives $m^2 = (x_1+x_2+x_3) \bmod p$ and hence $x_3 = (m^2 - x_1-x_2) \bmod p$. Then we compute y_3 using $y_3 = (mx_3+c) \bmod p$ and finally $P_1 + P_2 = (x_3, -y_3)$.

In Short: Addition algorithm for $P_1 = (x_1, y_1)$ and $P_2 = (x_2, y_2)$ on the elliptic curve E is:

1) If $P_1 \neq P_2$ and $x_1 = x_2$ then $P_1 + P_2 = \{O\}$.

2) If $P_1 = P_2$ and $y_1 = 0$ then $P_1 + P_2 = 2P_1 = \{O\}$.

3) If $P_1 \neq P_2$ (and $x_1 \neq x_2$), let $m = \frac{y_2-y_1}{x_2-x_1} \bmod p$ and $c = \frac{y_1x_2-y_2x_1}{x_2-x_1} \bmod p$.

4) If $P_1 = P_2$ and $y_1 \neq 0$, let $m = \frac{3{x_1}^2+a}{2y_1} \bmod p$ and $c = \frac{-x^3+ax+b}{2y} \bmod p$.

Then $P_1 + P_2 = ((m^2 - x_1 - x_2) \bmod p,\ (-m^3 + m(x_1 + x_2) - c) \bmod p)$. Scalar multiplication tP over E/F_p means $tP = P + P + ... + P(t \ \ times)$, that can be calculated using double-and-add method.

2.3 Complexity Assumption

Elliptic curve discrete logarithm problem (ECDLP): Given $x \in_R {Z_n}^*$ and P the generator of G and $Q \in G$, to compute x s.t. $Q = xP$ is called ECDLP and assumed to be intractable.

3 Formal Models of Multi Proxy Signature Scheme

The proposed model involves three parties: the original signer A, a set of l proxy signers $L = \{PS_1, PS_2, .., PS_l\}$, and a verifier. One of the proxy signers plays the role of clerk who combines all the partial proxy signatures and generates an MPS.

3.1 Definition of MPS Scheme

A MPS scheme is specified by the following polynomial-time algorithms.

- Setup: Given a security parameter k, this algorithm outputs the system parameters.
- Extract: It takes the security parameter k and outputs the secret-public key pair (sk_U, pk_U), for each user U participating in the scheme.
- DelGen: Given the systems' parameter, the original signer's private key and the warrant m_w to be signed, this algorithm outputs the delegation $W_{A\rightarrow PS_i}, (1 \leq i \leq l)$.
- DelVerif: The delegation verification algorithm, takes the original signer's public key, delegation $W_{A\rightarrow PS_i}, (1 \leq i \leq l)$ as inputs and verifies whether it is a valid delegation came from A.
- PKGen: The proxy key generation algorithm, takes $W_{A\rightarrow PS_i}, (1 \leq i \leq l)$ and some other secret information (for example, the secret keys of the executors) as inputs, and outputs a proxy signing key psk_{P_i} $\forall 1 \leq i \leq l$ for proxy signature.
- MPSign: The proxy signing algorithm, takes the proxy signing keys $psk_{P_i}, \forall 1 \leq i \leq l$ of all proxy signers and a message $m \in \{0,1\}^*$ as inputs, and outputs an MPS signature on behalf of A.
- MPVerif: The proxy verification algorithm, takes public keys of original signer, all proxy signers, and a proxy signature (m_w, σ, m, S) as inputs, and outputs 0 or 1. In the later case, (m, S) is a valid MPS for m by the proxy group L on behalf of the original signer A.

3.2 Security Model of MPS Scheme

We define the security of our MPS scheme under existential unforgeability against adaptive chosen message attack (EUF-ACMA) [3]. The security notion is based on the following game played between a challenger C and a probabilistic polynomial time adversary T under an experiment Exp_T^{MPS} of the adversary T.

- Setup: The challenger C runs this algorithm with input k and generates the public parameters. In addition C runs the Extract algorithm to obtain a public key pk and private key sk. The adversary T is given pk and system parameters while sk is kept secret.
- Queries. T can make the following queries adaptively to C.
  1) DelGen-query: T requests for the delegations with at-most q_s no of message warrant's for proxy signers with pk_{PS_i}, $(i = 1, 2, ...l)$ on behalf of original signer with pk_A adaptively. There exist a simulator S that simulates the DelGen oracle and outputs the corresponding valid delegations $W_{A\rightarrow PS_i}$ for each query.
  2) MPSign-query: T queries the signature oracle for at-most q_s no of messages under the obtained delegation $W_{A\rightarrow PS_i}$. There exist a simulator S that simulates the MPSign oracle and outputs the valid signature tuples.
- Output: Eventually, T outputs a tuple $(m_w,\ pk_A,\ pk_ps_i,\ \sigma,\ m,\ S)$, $(i = 1, 2, , l)$ and wins the game i.e Exp_T^{MPS} returns yes if
  1) Message warrant m_w and message m are not queried before for delegation and signature respectively.
  2) DelVerif $(pk_A, PK_ps_i; m_w, sigma) = valid$.
  3) MPVerif $(pk_A, PK_ps_i, m_w, sigma, m, S) = valid$.
  4) Otherwise returns No.

An MPS scheme is said to be existential delegation and signature unforgeable against adaptive chosen message attack (DS-EUF-ACMA), if for any polynomial-time adversary T, $Pr[Exp_T^{MPS}(k) = yes]$ is negligible.

4 Proposed Scheme

In this section, we present an MPS scheme without pairings. The proposed scheme involves three parties: the original signer A, a set of l proxy signers $L = \{PS_1, PS_2, .., PS_l\}$, and a verifier. One of the proxy signers plays the role of clerk who combines all the partial proxy signatures and generates an MPS on message m which confirms the warrant m_w. Our scheme mainly consists of the following seven algorithms.

- Setup: Takes a security parameter k, and returns the system parameters $\Omega = \{F_p, E/F_p, G, P, H_1, H_2, \}$ as defined in 2.1. $H_1 : \{0,1\}^* \times G \rightarrow Z_n^*$ and $H_2 : \{0,1\}^* \times G \rightarrow Z_p^*$ are two cryptographic secure hash functions.
- Extract: Each participant U of the scheme picks at random $sk_U \in Z_n^*$ and computes $pk_U = sk_U P$. Thus (sk_U, pk_U) is the (secret, public) key pair of user U.
- DelGen: This algorithm takes A's secret key sk_A and a warrant m_w as inputs, and outputs the delegation $W_{A\rightarrow PS_i}$, $1 \leq i \leq l$ as follows:
  1) Generates a random $a \in Z_n^*$, computes $K = aP$.
  2) Computes $h_{i_A} = H_2(m_w, K, pk_{PS_i})$, $h_A = \sum_1^l h_{i_A}$ and $\sigma = (h_A sk_A + a) \bmod n$.

  A sends delegation $W_{A\rightarrow PS_i} = \{pk_A, pk_{PS_i}, m_w, K, \sigma\}$ to each proxy signer PS_i, $1 \leq i \leq l$.

- DelVerif: To verify the delegation $W_{A\to PS_i}$ on warrant m_w, each proxy signer PS_i first computes $h_{i_A} = H_2(m_w, K, pk_{PS_i})$ and $h_A = \sum_1^l h_{i_A}$, then checks whether $\sigma P = h_A pk_A + K$ holds. Accepts if it is equal, otherwise rejects.

- PKGen: If PS_i accepts the delegation $W_{A\to PS_i}$, he computes the proxy signing key psk_{PS_i}, $1 \le i \le l$ as follows: $psk_{PS_i} = (\sigma h_p + sk_{PS_i}) \bmod n$, where $h_p = H_1(m_w, pk_A, K)$. Using psk_{PS_i}, these proxy signers can cooperate to sign any message m which confirms to m_w on behalf of the original signer A.

- MPSign: Each proxy signer $PS_i, (1 \le i \le l)$ chooses $a_i \in Z_n^*$, computes $N_i = a_i P$ and broadcasts his N_i to the other $l-1$ proxy signers. Then each PS_i computes $S_{PS_i} = (psk_{PS_i} + a_i h) \bmod n$ where $h = H_2(m, N)$, $N = \sum_1^l N_i$ and sends $(pk_A, pk_{PS_i}, K, m_w, m, S_{PS_i})$, $(1 \le i \le l, i \ne j,$ if PS_j is designated as clerk) to the clerk as his partial proxy signature. The clerk verifies the partial proxy signatures by checking the equation $S_{PS_i}P = h_p(h_A pk_A + K) + pk_{PS_i} + hN_i$, where $N = \sum_1^l N_i$, $h = H_2(m, N)$, $h_A = \sum_1^l H_2(m_w, K, pk_{PS_i})$ and $h_p = H_1(m_w, pk_A, K)$. If it holds, then he combines $S = \sum_1^l S_{PS_i}$ and sends the tuple $(pk_A, pk_{PS_i}, K, N, m_w, m, S), \forall 1 \le i \le l$ to verifier.

- MPVerif: To verify the signature (pk_A, pk_{PS_i}, K, N, m_w, m, S), $\forall 1 \le i \le l$ for message m, the verifier does as follows.

  Checks whether the message m confirms to the warrant m_w. If not, stop. Otherwise, continue. Checks whether the l proxy signers are authorized by the original signer in the warrant m_w. If not, stop. Otherwise, continue. Computes $h_{i_A} = H_2(m_w, K, pk_{PS_i})$, $h_A = \sum_1^l h_{i_A}$, $h_p = H_1(m_w, pk_A, K)$ and $h = H_2(m, N)$, then checks whether the equation: $SP = lh_p(h_A pk_A + K) + \sum_1^l pk_{PS_i} + hN$ holds. If holds then accepts otherwise rejects it.

Correctness. Since, $S_{PS_i}P = h_p(h_A pk_A + K) + pk_{PS_i} + hN_i$ and $N = \sum_1^l N_i$, we have,

$$\begin{aligned} SP &= \sum_1^l S_{PS_i}P \\ &= \sum_1^l [h_p(h_A pk_A + K) + pk_{PS_i} + hN_i] \\ &= lh_p(h_A pk_A + K) + \sum_1^l pk_{PS_i} + hN. \end{aligned}$$

5 Security Analysis

In this section, we will examine the security of our proposed scheme. Assume there is an adversary T who can break our proxy signature scheme (say $\sum$). We will construct a polynomial-time algorithm F that, by simulating the challenger C and interacting with T, solves the ECDLP.

Theorem 1. *Consider an adaptively chosen message attack in the random oracle model(ROM) against $\sum$. If there is an attacker T that can break $\sum$ with at most q_{H_2} H_2-queries and q_s signature queries within time bound t and non negligible probability ε. Then there exist an algorithm that solves ECDLP with non-negligible probability.*

Proof. Suppose an attacker T can break $\sum$ through adaptively chosen message attack then $Pr[Exp_T^{MPS}(k) = yes]$ is non negligible. We will show that using the ability of T and forking lemma [15], an algorithm F can be constructed for solving the ECDLP. Forking reduction technique works because the challenger sets the random oracle answers so that one set of questions from adversary are answered with a number of completely independent sets of answers.

For this purpose F sets $\{F_p, E/F_p, G, P, P_{pub}, H_1, H_2\}$ as system parameters and answers T's queries 3.2 as follows.

Case 1. (*Existential Delegation Unforgeable under Adaptive Chosen Message Attack*). The challenger C interacts with forger T and responds as follows.

- Setup: C starts to obtain public key pk and private key sk. The adversary T is given pk.

- DelGen-query: T is allowed to query the delegation oracle for $m_w, pk_A, pk_{PS_i}, \forall 1 \le i \le l$. There exist a simulator S that simulates the oracle and outputs (σ, K) that satisfies the equation $\sigma P = h_A pk_A + K$. Thus σ is a valid signature on m_w for pk_A.

- Output: If T can forge a valid delegation on warrant m_w without knowing the secret key with the probability $Pr[Exp_T^{MPS}(k) = yes] = \varepsilon \ge 10(q_{H_2}+1)(q_{H_2}+q_s)/2^k$ where m_w has not been queried to the delegation oracle (as Lemma 4 of [15] aims), then a replay of F with the same random tape but different choice of H_2 will output two valid delegations $\{pk_A, pk_{PS_i}, m_w, K, \sigma, h_A\}$ and $\{pk_A, pk_{PS_i}, m_w, K, \sigma', h_A'\}$.

Then we have

$$\sigma P = h_A pk_A + K \quad (1)$$

$$\sigma' P = h_A' pk_A + K. \quad (2)$$

From Equations (1) and (2), we have

$$(\sigma - \sigma')P = (h_A - h_A')sk_A P.$$

Let $u = \sigma - \sigma'$ and $v = (h_A - h_A')^{-1}$, then

$$sk_A = uv \bmod n.$$

Table 1: Cryptographic operation time (in milliseconds)

Operation	Modular exp.	O_P	M_P	M_E	H_M	General hash
Time	5.31	20.04	6.38	2.21	3.04	< 0.001

Table 2: Computational cost comparison

Scheme	Extract	DelGen	DelVerif	PKGen	MPSign	MPVerif	Total
Scheme [10]	$1M_P$	$1M_P + 1H_M$	$1H_M + 2O_P$	$1M_p + 1H_M$	$2M_P + 1H_M + 3O_p$	$1H_M + 2O_P$	$5M_P + 5H_M + 7O_P$
Scheme [23]	$1M_P$	$1M_P + 1H_M$	$1H_M + 2O_P$	$1M_p + 1H_M$	$2M_P + 1H_M + 3O_p$	$2M_P + 1H_M + 2O_P$	$6M_P + 5H_M + 7O_P$
Our scheme	$1M_E$	$1M_E$	$2M_E$	$0M_E$	$5M_E$	$4M_E$	$13M_E$

Table 3: Running time comparison (in ms)

Scheme	Extract	DelGen	DelVerif	PKGen	MPSign	MPVerif	Total
Scheme [10]	9.42	9.42	49.50	6.38	61.36	49.50	185.58
Scheme [23]	9.42	19.14	46.46	6.38	50.74	58.92	191.06
Our scheme	2.21	2.21	6.63	≈ 0	15.47	13.26	39.78

According to Lemma 4 [15] the ECDLP can be solved with probability $\varepsilon^{'} \geq 1/9$ and time $t^{'} \leq 23\ q_{H_2} t/\varepsilon$.

Case 2. (*Existential Signature Unforgeable under Adaptive Chosen Message Attack*). From Case 1, it is clear that the adversary T can not generate a valid delegation. In this Case the challenger C interacts with forger T as follows.

- Setup: C starts to obtain public key pk and private key sk. The adversary T is given pk.
- MPSign-query: T is allowed to query the signature oracle for m under the delegation $W_{A \to PS_i} = \{pk_A, pk_{PS_i}, m_w, K, \sigma\}$. There exist a simulator S that simulates the oracle and generates a tuple (N, S) that satisfies the equation $SP = lh_p(h_A pk_A + K) + \sum_1^l pk_{PS_i} + hN$.
- Output: If T can forge a valid signature on message m with the probability $Pr[Exp_T^{MPS}(k) = yes] = \varepsilon \geq 10(q_{H_2} + 1)(q_{H_2} + q_s)/2^k$ where m has not been queried to the signature oracle, then a replay of F four times with the same random response but different choices of H_2, will output four valid signatures $(pk_A, pk_{PS_i}, K, N, m_w, m, S^j, h_A^j, h^j)$, $\forall 1 \leq i \leq l$ and $j = 1, 2, 3, 4$.

Then we have

$$S^j P = lh_p(h_A^j pk_A + K) + \sum_1^l pk_{PS_i} + h^j N. \qquad (3)$$

If sk_A, a, b, y denote elliptic curve discrete logarithms of $pk_A, K, \sum_1^l pk_{PS_i}$ and N respectively. Then from equation (3), we have

$$S^j = lh_p(h_A^j sk_A + a) + b + h^j y, \quad j = 1, 2, 3, 4.$$

Since, in the above four equations, the unknowns sk_A, a, b, y neither have any power nor multiplied together. So these equations are linear. We consider that with high probability the determinant of the system obtained by the above four linear equations is non zero and so these equations are linearly independent.

Therefore, there exist an algorithm F that solves the above four linearly independent equations, and outputs sk_A as the solution of the ECDLP with probability $\varepsilon^{'} \geq 1/9$ and time $t^{'} \leq 23 q_{H_2} t/\varepsilon$ (Lemma 4 [15]). □

6 Comparative Analysis

In this section, we will compare the efficiency of our scheme with the schemes [10, 23]. We use the running time of different cryptographic operations calculated by [6] in some cryptographic environment for such efficiency comparison as given in Table 1.

Where M_E, M_P, H_M, O_P stand for one ECC based scalar multiplication, pairing based scalar multiplication, Map-to-point hash function and pairing operation respectively.

Computational cost and running time analysis of our scheme with schemes [10, 23] are given in Tables 2 and 3, respectively.

From the above Table 2, it is clear that the running time of MPSign algorithm of our scheme is 14.54% of scheme [10] as well as of scheme [23]. Total running time of our scheme is 20.50% of scheme [10] and 17.85% of the

scheme [23].

Note: Although our proposed scheme is based on ECC, it does not use pairings. Therefore one can easily conclude by efficiency comparison that our proposal is much more efficient than other existing MPS schemes from pairings.

7 Conclusion

In this paper, we proposed an efficient provable secure multi-proxy signature scheme based on ECC without using pairings that also avoids the map-to-point hash function. For this proposal, we first defined a model and then proved the security of proposed scheme against adaptive chosen message attack under ECDL-assumption. Compared with previous schemes, the new scheme reduces the running time of signing algorithms heavily. Therefore, our scheme is more efficient and applicable than the previous related schemes in practice.

References

[1] L. Chen, Z. Cheng, and N. P. Smart, "Identity-based key agreement protocols from pairings," *International Journal of Information Security*, vol. 6, pp. 213–241, 2007.

[2] C. Feng and C. Zhenfu, "A secure identity-based multi-proxy signature scheme," *Computer and Electrical Engineering*, vol. 35, pp. 86–95, 2009.

[3] S. Goldwasser, S. Micali, and R. L. Rivest, "A digital signature scheme secure against adaptive chosenmessage attacks," *SIAM Journal of Computing*, vol. 17, no. 2, pp. 281–308, 1988.

[4] C. Gu and Y. Zhu, "Provable security of ID-based proxy signature schemes," in *Networking and Mobile Computing*, LNCS 3619, pp. 1277–1286, Springer-Verlag, 2005.

[5] C. Gu and Y. Zhu, "An efficient ID-based proxy signature scheme from pairings," in *Information Security and Cryptology*, LNCS 4990, pp. 40–50, Springer-Verlag, 2008.

[6] D. He, J. Chen, and J. Hu, "An ID-Based proxy signature schemes without bilinear pairings," *Annalas of Telicommunications*, vol. 66, no. 11-12, pp. 657–662, 2011.

[7] S. J. Hwang and C. C. Chen, "New multi-proxy multi-signature schemes," *Appllied Mathematics and Computation*, vol. 147, pp. 257–67, 2004.

[8] H. Ji, W. Han, and L. Zhao et al, "An identity-based proxy signature from bilinear pairings," in *Proceedings of WASE International Conference on Information Engineering*, pp. 14–17, 2011.

[9] N. Koblitz, "Elliptic curve cryptosystems," *Mathenmatics of Computation*, vol. 48, no. 177, pp. 203–209, 1987.

[10] S. Li and F. Zhang, "A new Multi-Proxy signature from bilinear pairing," *Journal of Electronics (China)*, vol. 24, no. 1, pp. 90–94, 2007.

[11] Z. Liu, Y. Hub, X. Zhang, and H. Maa, "Provably secure multi-proxy signature scheme with revocation in the standard model," *Computer Communication*, vol. 34, pp. 494–501, 2011.

[12] M. Lochter, J. Merkle, *Elliptic Curve Cryptography (ECC) Brainpool Standard Curves and Curve Generation*, RFC 5639, Mar. 2010.

[13] M. Mambo, K. Usuda, and E. Okamoto, "Proxy signatures: Delegation of the power to sign messages," *IEICE Transaction Fundamentals*, vol. E79-A(9), pp. 1338–1353, 1996.

[14] V. Miller, "Uses of elliptic curves in Cryptography," in *Proceedings of Advances in Cryptology-Crypto 85*, pp. 417–426, Santa Barbara, USA, Aug. 1985.

[15] D. Pointcheval and S. Jacque, "Security arguments for digital signatures and blind signatures," *Journla of Cryptology*, vol. 13, no. 3, pp. 361–396, 2000.

[16] N. Tiwari and S. Padhye, "An ID-Based proxy multi signature scheme without bilinear pairings," in *First International Conference on Security Aspects in Information Technology (InfoSecHiComNet '11)*, pp. 83–92, Haldia, India, Oct. 2011.

[17] N. Tiwari and S. Padhye, "Analysis on the generalization of proxy signature," *Security and Communication Network*, vol. 6, pp. 549–556, 2013.

[18] N. Tiwari, S. Padhye, and D. He, "Efficient ID-based multi-proxy multi-signature without bilinear maps in ROM," *Annalas of Telecommunication*, vol. 68, no. 3-4, pp. 231–137, 2013.

[19] A. Wang, J. Li, and Z. Wang, "A provably secure proxy signature scheme from bilinear pairings," *Journal of Electronnics (China)*, vol. 27, no. 3, pp. 298–304, 2010.

[20] T. S. Wu, C. L. Hsu, and H. Y. Lin, "Self-certified multi-proxy signature schemes with message recovery," *Jorunal of Zhejiang University Science*, vol. 10, no. 2, pp. 290–300, 2009.

[21] W. Wu, Y. Mu, and W. Susilo et al., "Identity-based proxy signature from pairings," in *The 4th International Conference, Authentic and Trusted Computing (ATC'07)*, pp. 22–31, Hong Kong, China, July 2007.

[22] L. Xiangxue and C. Kefei, "Id-based multi-proxy signature, proxy multi-signature and multi-proxy multi-signature schemes from bilinear pairings," *Appllied Mathematics and Computation*, vol. 169, pp. 437–450, 2005.

[23] L. Xiangxue, C. Kefei, and L. Shiqun, "Multi-proxy signature and proxy Multi-signature schemes from bilinear pairings," in *Parallel and Distributed Computing: Applications and Technologies (PDCAT'04)*, pp. 591–595, Singapore, Dec. 2004.

[24] C. Xiaofeng, Z. Fangguo, and K. Kwangjo, "ID-Based Multi-Proxy signature and blind multisignature from bilinear pairings," in *Proceeding of KIISC Conference*, p. 1119, Korea, Nov. 2003.

[25] Q. Xue and Z. Cao, "Improvement of multi-proxy signature scheme," in *Proceeding of IEEE Fourth International Conference on Computer and Information Technology*, pp. 450–455, Sep. 2004.

[26] L. J. Yi, G. Q. Bai, and G. Z. Xiao, "Proxy multisignature scheme: A new type of proxy signature scheme," *Electronics Letters*, vol. 36, no. 6, pp. 527–528, 2000.

[27] J. Zhang and W. Zou, "Another ID-based proxy signature scheme and its extension," *Wuhan University Journal of Natural Science*, vol. 12, pp. 133–136, 2007.

Namita Tiwari received her B.Sc. degree from C. S.J.M. University, Kanpur, India in 2006 and M.Sc. degree in Mathematics from Indian Institute of Technology, Kanpur, India in 2008. She did her Ph.D. from Motilal Nehru National Institute of Technology, Allahabad, India in 2013. She is a life member of Cryptology Research Society of India (CRSI). Her area of interest is Digital Signature.

Sahadeo Padhye received his B.Sc.and M.Sc. degree in Mathematics form Pt. Ravishankar Shukla University, Raipur. Chhattisgarh, India in 1999 and 2001. Council of Scientific and Industrial Research (CSIR), India has granted him Junior Research Fellowship (2002-2004). He did his Ph.D. form Pt. Ravishankar Shukla University, Raipur, India. He is a life member of Cryptology Research Society of India (CRSI)and Indian Mathematical Society and a member of International Association of Cryptologic Research (IACR). His area of interest is Public Key Cryptography based on elliptic curve and digital signature. Presently he is working as Assistant Professor in the Department of Mathematics, Motilal Nehru National Institute of Technology, Allahabad, India.

Refereed Computation Delegation of Private Sequence Comparison in Cloud Computing

Xu Ma[1], Jin Li[2], and Fangguo Zhang[1]
(Corresponding author: Fangguo Zhang)
School of Information Science and Technology, Sun Yat-Sen University[1]
Guangzhou 510006, P. R. China
School of computer science, Guangzhou University[2]
Guangzhou 510006, P. R. China
(Email: xumasysu@gmail.com, lijin@gzhu.edu.cn, isszhfg@mail.sysu.edu.cn)
(Received June 24, 2013; revised and accepted Sept. 28 & Nov. 26, 2013)

Abstract

Sequence comparison has been widely used in many engineering systems, such as fuzzy keyword search, plagiarism detection, and comparison of gene sequences. However, when the length of the string is extraordinarily long, like the DNA sequence that contains millions of nucleotides, sequence comparison becomes an intractable work, especially when the DNA database is big and the computation resources are limited. Although the generic computation delegation schemes provide a theoretically feasible solution to this problem, it suffers from severe inefficiency when we directly substitute the general function by the sequence comparison function. In this paper, we focus on refereed computation delegation of sequence comparison and present the refereed computation delegation scheme of sequence comparison using multiple servers. In our scheme, the user can detect the dishonest servers and get the correct answer as long as there is one honest server. The direct application of our scheme is DNA sequence comparison of big gene database in medical system. Meanwhile, our scheme satisfies the security requirement of sequence privacy against the malicious adversaries. Moreover, since neither the fully homomorphic encryption nor the complicated proof systems are used for the problem generation and result verification, our solution clearly outperforms the existing schemes in terms of efficiency. The computation complexity of the user is reduced from $O(mn)$ to $O(\log^2(mn))$, where m,n are the length of the sequences.

Keywords: Privacy, refereed computation delegation, sequence comparison

1 Introduction

Sequence comparison can be viewed as the string editing problem, i.e., computing the distance between two strings. The edit distance is one of the most widely used notions of similarity: it is the least-cost operation set of deletions, insertions and substitutions required to transform one string into another. Sequence comparison is widely used in many engineering systems, such as fuzzy keyword search [11, 28], plagiarism detection, and comparison of gene sequences [9]. However, the computation complexity of sequence comparison is $O(mn)$, where m and n are the respective length of the strings. When the length of the string is extraordinarily long, like the DNA sequence that contains millions of nucleotides, sequence comparison becomes an intractable work, especially for resource limited devices.

Generally, such computation expensive tasks can deploy the so-called "computation delegation" to accomplish the tasks efficiently. Computation delegation [1, 15, 16] considers a scenario where one party, the delegator who is computationally weak, wishes to delegate the computation of a function f on various inputs $x_1, x_2, \cdots, x_k$ to one or more servers who are computationally strong. However, the servers are not fully trusted, the basic security requirements of computation delegation are verifiability and efficiency, which require that the delegator should be able to verify the correctness of the values returned by the worker. Moreover, the verification process should require substantially less computation efforts than computing $f(x)$ from scratch. In addition, an important property of verifiable computation is privacy, which enables the delegator to hide some private information from the worker. As we know, individual DNA and protein sequences are highly sensitive and vulnerable to re-identification even when anonymized. Therefore, the outsourcing technique should enable the desired computation without revealing any information about the sequences to the parties carrying out the computation.

However, if we trivially apply the traditional computation delegation scheme to sequence comparison, the resulting scheme becomes severely inefficient. The reason

is that most of the previous works are based on complex cryptographic tools, such as fully homomorphic encryption [17] and proof systems [6, 19].

In [4], Atallah et al. proposed a secure outsourcing scheme of sequence comparison. However, the protocol was only proved to be privacy secure in semi-honest adversary model, and the most significant security requirement of verifiability was not satisfied. Apart from [4], most of the previous works related to sequence comparison have been done in the framework of two-party computation model [10], in which two parties with private inputs wish to jointly compute some function of their inputs while preserving certain properties like privacy and correctness. These works are quite related to private pattern matching, which is out of the scope of this paper.

Our contributions. In this paper, we present a new computation delegation model, which is called refereed computation delegation of sequence comparison, using multiple servers. Our contributions are two-fold:

- The user can detect the dishonest servers and get the correct answer as long as there is at least one honest server.
- Our scheme satisfies the security requirement of input/output privacy against the malicious servers.

In multi-server model, one trivial method to realize the verifiability is: the user picks N different servers and asks each of those to execute his programme and return the output. Now, the user takes the plurality value of those answers to be the correct answer. As long as there is a majority of honest servers, the user gets the correct answer. The main drawback of this approach is the need for an honest majority of servers.

To get better performance and abate the assumption of a plurality of honest servers, we propose a new approach which is called refereed computation delegation of sequence comparison based on Canetti's computation delegation scheme [12]. In our scheme, the user runs like a referee who supervises the servers. The servers do the computation and return the result together with a commitment of the result back to the user. In the verification process, when the user detects inconsistency between the returned results, the process of consistency proof and verification are activated. After that, the user can get the correct result by performing only a *single* step of the computation of sequence comparison. Specially, our scheme is implementable suppose that there are only two servers, one of which is honest.

As for efficiency, since neither the fully homomorphic encryption nor the complicated proof systems are used for the problem generation and result verification, our solution clearly outperforms the existing schemes in terms of efficiency. In detail, the computation load of a server is $O(cmn)$, and the computation complexity of the user is reduced from $O(mn)$ to $O(\log^2(mn))$, where m,n are the length of the sequences and c is a constant.

Related work. Atallah et al. proposed a secure outsourcing scheme of sequence comparison in [4] using two non-collusion servers. However, the security model of the protocol is semi-honest and the security requirement of verifiability was not mentioned. In [14,24,38], the authors studied the sequence comparison in the framework of two-party computation model, in which two parties with private inputs wish to jointly compute some function of their inputs while preserving the security requirements like privacy and correctness. These works are quite related to private pattern matching [14, 21, 22, 39], where party P_1 holds a pattern and party P_2 holds a text. The goal of P_1 is to learn where the pattern appears in the text, without revealing it to P_2 and learning anything else about P_2's text.

Computation delegation has received widespread attention due to the rise of cloud computing [13,37], where businesses buy computing power from a service, rather than purchasing and maintaining their own computing resources. Another motivation of computation delegation is the proliferation of mobile devices, such as netbooks and smart phones. Due to the computation and storage limitations, sometimes it is desirable to off-load heavy computations, such as cryptographic operations, or photo manipulation, to the cloud server. However, the cloud server is not fully trusted and sometimes the applications outsourced to the cloud are so critical that it is imperative to keep the original data private and rule out accidental errors during the computation.

Previous research on computation delegation can be classified into two categories: 1) the generic computation delegation [1, 15, 16]: it can be applied for arbitrary functions; 2) the concrete computation delegation [2, 5, 7, 23, 25, 29, 30]: they are designed for some specific functions, such as polynomial evaluation and linear algebra. In the generic model, most of the previous works are based on fully homomorphic encryption [17] and proof systems, such as interactive proofs [6, 19], efficient arguments based on probabilistically checkable proofs (PCP) [26, 27], CS proofs [32]and the muggles proofs in [18]. The complex cryptographic tools used in the generic model result in inefficiency when applying these protocols to some concrete functions.

For the computation delegation of specific functions, plenty of research works have been proposed. Benjamin and Atallah [8] addressed the problem of secure outsourcing for widely applicable linear algebra computations. However, the proposed protocols required the expensive operations of homomorphic encryptions. Atallah and Frikken [1] further studied this problem and gave improved protocols based on the so-called weak secret hiding assumption. Benabbas et al. [7] presented the first practical computation delegation scheme for high degree polynomial functions based on the approach of [16]. In 2011, Green et al. [20] proposed new methods for efficiently and securely outsourcing decryption of attribute-based encryption (ABE) ciphertexts. Based on this work, Parno et al. [35] showed a construction of a multi-function computation delegation scheme.

Organization. The paper is organized as follows. In

Section 2, we give a brief description of the preliminaries which will be used in the following sections, In Section 3, we present the system model and security definition of our scheme. The construction of our scheme is presented in Section 4. In Section 5, we give the efficiency analysis and security proof of our scheme. Finally, we conclude in Section 6.

2 Preliminaries and Tools

2.1 Edit Distance

We now precisely give the definition of edit distance [40]. Consider a finite alphabet set Σ whose elements will be used to construct strings. Let $\mathbb{C}_I$, $\mathbb{C}_D$, $\mathbb{C}_S$ be finite sets whose elements are finite integers. And let the function $I : \Sigma \to \mathbb{C}_I$ be insertion cost function, i.e., $I(a)$ is the cost of inserting an element $a \in \Sigma$ to a given string. Similarly, define the deletion cost function as $D : \Sigma \to \mathbb{C}_D$, i.e., $D(a)$ is the cost of deleting an element $a \in \Sigma$ from a given string. And define the substitution cost function as $S : \Sigma \times \Sigma \to \mathbb{C}_S$, i.e., $S(a, b)$ is the cost of replacing an element $a \in \Sigma$ in a given string by an element $b \in \Sigma$.

If we let λ be a string of length n, $\lambda = \lambda_1, \lambda_2, \cdots, \lambda_n$, and μ be a string of length m, $\mu = \mu_1, \mu_2, \cdots, \mu_m$, both are strings over alphabet set Σ. As mentioned above, there are three allowed edit operations to be operated on λ, insertion of an element, deletion of an element and substitution of one element by another. Each sequence of operations that transforms λ into μ has an aggregate cost associated with it, which is equal to the sum of the costs of the operations in it. The least-cost of such sequences is the edit distance.

We now give a brief review of the standard dynamic programming for computing edit distance [40]. Let $M(i, j)$ $(0 \le i \le n, 0 \le j \le m)$be the minimum cost of transforming the prefix of λ of length i into the prefix of μ of length j, i.e., of transforming $\lambda_1, \lambda_2, \cdots, \lambda_i$ into $\mu_1, \mu_2, \cdots, \mu_j$. Then $M(0, 0) = 0$, $M(0, j) = \sum_{k=0}^{j} I(\mu_k)$ for $1 \le j \le m$, and $M(i, 0) = \sum_{k=0}^{i} D(\lambda_k)$, for $0 \le i \le n$. For positive i and j, we have

$$M(i,j) = min \begin{cases} M(i-1, j-1) + S(\lambda_i, \mu_j) \\ M(i-1, j) + D(\lambda_i) \\ M(i, j-1) + I(\mu_j) \end{cases}$$

for all $1 \le i \le n$, $1 \le j \le m$. Hence, $M(i, j)$ can be evaluated row by row or column by column in $O(mn)$. Observe that, of all the entries of the M-matrix, only the three entries $M(i-1, j-1)$, $M(i-1, j)$ and $M(i, j-1)$ are involved in the computation of the final value $M(i, j)$.

2.2 Merkle Hash Tree

Merkel Hash Tree (MHT) [31] is a common primitive that allows one to hash a long string of n characters in a way that the hash can later be used to reveal any part of the string and supply a short proof of consistency. The construction of MHT is based on the collision-resistant hash function, and given a collision-resistant hash function H and string str of length n, the tree has n leaf nodes where leaf node i has the value of $H(str[i])$, $str[i]$ is the i-th character of str. The next level has the values of $H(H(str[i])||H(str[i+1]))$, for $i = 1, 3, \cdots, n-1$, and so on for the other levels. The proof of consistency for character i consists of $H(str[i])$ and all the sibling hash values of the nodes along the path from the root to the leaf node $H(str[i])$.

Given a MHT of a string str, denote by $MH_{root}(str)$ the value of the root, by $MH_{proof}(str, i)$ the proof of consistency for the i-th character, and by $VerMHP(root, i, str_i, p)$ the verification function that given a claimed proof $p = MH_{proof}(str, i))$ outputs true if p is valid and false otherwise. Note that the size of the proof is $\log n$ and the complexity of verification function is $O(\log n)$.

3 System Model and Security Definitions

3.1 System Model

A refereed computation delegation $\mathcal{RCD}$ [12] for a function f is a protocol between a user (or referee) R and N servers $S_1, S_2, \cdots, S_N$. The difference between traditional computation delegation model and $\mathcal{RCD}$ is that the user acts like a referee in the delegation procedure. The user is able to detect the dishonest server when there is a dispute. The user and the servers receive the input x. The servers compute the function in parallel and claim different result of the computation of $f(x)$, and the user should be able to detect the dishonest servers and determine the correct $f(x)$ with overwhelming probability as long as there is at least one honest server. Formally, for any input x and for all $i \in \{1, \cdots, N\}$, if S_i is honest, then for any potentially dishonest $S_1^*, \cdots, S_{i-1}^*, S_{i+1}^*, \cdots, S_N^*$, the output of R is $f(x)$ with probability at least $1 - \varepsilon$, where ε is negligible. An optional security requirement of $\mathcal{RCD}$ is I/O privacy. In the following subsection, we will give a formal definitions of the security requirements.

Firstly, we retrospect the traditional verifiable computation model. In detail, the traditional verifiable computation scheme consists of four algorithms defined below (KeyGen,ProbGen, Compute, Verify):

- $(pk, sk) \leftarrow$ KeyGen(f,λ): *Based on the security parameter λ, the randomized key generation algorithm generates a public key pk that encodes the target function $f(\cdot)$, which is used by the cloud server to compute $f(\cdot)$. It also computes a matching secret key sk, which is kept secret by the user U.*

- $(\sigma_x, \tau_x) \leftarrow \mathsf{ProbGen}_{sk}(x)$: *The problem generation algorithm uses the secret key sk to encode the input x as a public value σ_x, which is given to the cloud server*

to compute with, and a secret value τ_x, which is kept private by the user U.

- $\sigma_y \leftarrow \mathsf{Compute}_{pk}(\sigma_x)$: *Using the user's public key pk and the encoded input σ_x, the server computes and outputs an encoded version of the result $y = f(x)$.*

- $y \vee \perp \leftarrow \mathsf{Verify}_{sk}(\sigma_y, \tau_x)$: *Using the secret key sk and the secret "decoding" value τ_x, the verification algorithm converts the cloud server's encoded output into the output of $y = f(x)$ or $\perp$ indicating that σ_y does not represent the valid output of $f(\cdot)$ on x.*

Now we give a formal description of refereed computation delegation. In detail, a refereed computation delegation scheme $\mathcal{RCD}$=(KeyGen, ProbGen, Compute, Verify) consists of four algorithms as defined below, which is the same as that in the verifiable computation [7,16], except that now N servers receive the same input from the user and return the results to the user.

- $(pk, sk) \leftarrow \mathsf{KeyGen}(\mathrm{f}, \kappa)$: *Based on the security parameter κ, the randomized key generation algorithm generates a public key/secret key pair pk/sk for the function $f(\cdot)$. The public key is provided to the servers, while the secret key is kept private by the user.*

- $(\sigma_x, \tau_x) \leftarrow \mathsf{ProbGen}_{sk}(x)$: *The problem generation algorithm uses the secret key sk to encode the input x as a public value σ_x, which is given to all the cloud servers to compute with, and a secret value τ_x, which is kept private by the user U.*

- $\sigma_y \leftarrow \mathsf{Compute}_{PK}(\sigma_x)$: *Using the user's public key pk and the encoded input σ_x, the servers compute and output an encoded version of the result $y = f(x)$, respectively.*

- $y \cup \perp \leftarrow \mathsf{Verify}_{sk}(\sigma_y)$: *Using the secret key sk and the secret "decoding" value τ_x, the verification algorithm converts the cloud servers' encoded output $y*$. Note that $y* \neq y$ if the server is dishonest. Then U verifies the correctness of $y*$ and obtains the correct result as long as there is at least one honest server.*

The basic efficiency requirement of a $\mathcal{RCD}$ scheme is that the time to encode the input and verify the output must be smaller than the time to compute the function from scratch, and the complexity of the servers is polynomial in the complexity of evaluating f. Formally, A $\mathcal{RCD}$ can be outsourced if it permits efficient problem generation and verification. This implies that for any input x and output σ_y, the time required for $\mathsf{ProbGen}_{sk}(x)$ plus the time required for $\mathsf{Verify}_{sk}(\tau_x, \sigma_y)$ is smaller than T, where T is the time to compute the function $f(x)$ from scratch.

3.2 Security Requirements

A refereed computation delegation scheme should be both correct and secure. Intuitively, a $\mathcal{RCD}$ scheme is correct if the user always outputs the correct result $f(x)$ as long as there is at least one honest cloud server. In the following experiments, A denotes the set of malicious cloud servers who are allowed to collude with each other, of which the size is at most $N-1$. Note that in the refereed delegation of computation, all the servers receive the same input, therefore, the oracle answer for an adversary set A just contains one answer. And the members of the adversary set A will output the same result in order to improve the probability of successful attack. Thus, the adversary set A can be viewed as one party.

Experiment $\mathbf{Exp}_A^C[\mathcal{RCD}, f, \kappa]$
$(pk, sk) \leftarrow \mathsf{KeyGen}(f, \kappa)$;
For $i = 1, \cdots, l = poly(\kappa)$
$x_i \leftarrow A(x_1, \sigma_{x_1}, \beta_1, \cdots, x_{i-1}, \sigma_{x_{i-1}}, \beta_{i-1})$;
$(\sigma_{x_i}, \tau_{x_i}) \leftarrow \mathsf{ProbGen}_{sk}(x_i)$;
$\sigma_{y_i} \leftarrow A(pk, x_1, \sigma_{x_1}, \beta_1, \cdots, x_{i-1}, \sigma_{x_{i-1}}, \beta_{x_{i-1}}, \sigma_{x_i})$;
$\beta_i = \mathsf{Verify}_{sk}(\tau_{x_i}, \sigma_{y_i})$;
$x \leftarrow A(pk, x_1, \sigma_{x_1}, \beta_1, \cdots, x_l, \sigma_{x_l}, \beta_l)$
$(\sigma_x, \tau_x) \leftarrow \mathsf{ProbGen}_{sk}(x)$;
$\sigma_y \leftarrow A(pk, x_1, \sigma_{x_1}, \beta_1, \cdots, x_l, \sigma_{x_l}, \beta_l, \sigma_x)$;
$\hat{y} \leftarrow \mathsf{Verify}_{sk}(\tau_x, \sigma_y)$;
$\hat{y} \neq f(x)$, output 1, else 0;

In the above experiment, the malicious servers are given oracle access to generate the encoding of multiple problem instances, and also oracle access to the result of the verification algorithm on arbitrary strings on those instances. The adversary succeeds if they convince the user to output wrong result for a given input value. Our goal is to make the adversary succeed only with negligible probability.

Correctness. For a refereed computation delegation scheme $\mathcal{RCD}$, we define the advantage of a set of adversaries A in the experiment above as:

$$Adv_A(\mathcal{RCD}, f, \kappa) = Pr[\mathbf{Exp}_A^C[\mathcal{RCD}, f, \kappa] = 1]$$

A refereed computation delegation scheme $\mathcal{RCD}$ is correct if for any function $f \in \mathcal{F}$, and for any adversary set A whose members run in probabilistic polynomial time,

$$Adv_A(\mathcal{RCD}, f, \kappa) \leq neg(\kappa)$$

where $neg(\cdot)$ is a negligible function of its input.

While the basic security requirements of correctness protects the integrity of $\mathcal{RCD}$, it is also desirable to protect the input and output of the refereed computation delegation scheme against the malicious servers. Below, we define the input/output privacy based on a typical indistinguisability argument that guarantees that no information about the inputs/outputs is leaked.

Intuitively, a refereed verifiable computation scheme is input private when the public outputs of the problem generation algorithm ProbGen over two different inputs are indistinguishable. In the following experiment, the

adversaries are allowed to request the encoding of any input they desires. The $\mathsf{PubProbGen}$ returns the public parameter σ_x to the adversary.

Experiment $\mathbf{Exp}_A^{IPriv}[\mathcal{RCD}, f, \kappa]$
$(pk, sk) \leftarrow \mathsf{KeyGen}(f, \kappa)$
$(x_0, x_1) \leftarrow A^{\mathsf{PubProbGen}(\cdot)}(pk)$
$(\sigma_0, \tau_0) \leftarrow \mathsf{ProbGen}_{sk}(x_0)$
$(\sigma_1, \tau_1) \leftarrow \mathsf{ProbGen}_{sk}(x_1)$
$b \leftarrow_R \{0, 1\}$
$\hat{b} \leftarrow A^{\mathsf{PubProbGen}_{sk}(\cdot)}(pk, x_0, x_1, \sigma_b)$
If $\hat{b} = b$, output1, else 0.

Input Privacy. For a refereed delegation of computation scheme, we define the advantage of an adversary set A in the experiment above as :

$$Adv_A^{IPriv}(\mathcal{RCD}, f, \kappa) = |Pr[\mathbf{Exp}_A^{IPriv}[\mathcal{RCD}, f, \kappa] = 1] - \frac{1}{2}|.$$

A refereed computation delegation scheme is input private for a function f, if for any adversary set A whose members run in probabilistic polynomial time,

$$Adv_A^{IPriv}(\mathcal{RCD}, f, \kappa) \leq neg(\kappa)$$

where $neg()$ is a negligible function of its input.

A similar definition can be made for output privacy. We consider the following the experiment. During the process, the adversaries are allowed to request the output of the algorithm $y \cup \perp \leftarrow \mathsf{Verify}$. In the challenged phase, the adversary has to submit two encoded outputs σ_0 and σ_1. After the adversaries have chosen the challenged output, they can continue the query process. The experiment ideally simulates the execution process of the refereed computation delegation scheme. The oracle $\mathsf{OVerify}$ returns the output of the algorithm $y \cup \perp \leftarrow \mathsf{Verify}$ to the adversaries.

Experiment $\mathbf{Exp}_A^{OPriv}[\mathcal{RCD}, f, \lambda]$
$(pk, sk) \leftarrow \mathsf{KeyGen}(f, \lambda)$
$(\sigma_{y_0}, \sigma_{y_1}) \leftarrow A^{\mathsf{OVerify}}_{\mathsf{PubProbGen}}(pk, f, \lambda)$
$y_0 \leftarrow \mathsf{Verify}(\sigma_{y_0}, f, \lambda)$
$y_1 \leftarrow \mathsf{Verify}(\sigma_{y_1}, f, \lambda)$
$b \leftarrow \{0, 1\}$
$\hat{b} \leftarrow A^{\mathsf{OVerify}}_{\mathsf{PubProbGen}}(f, pk, y_b)$
if $\hat{b} = b$, output 1, else 0.

Output Privacy. For a refereed computation delegation scheme $\mathcal{RCD}$, we define the advantage of an adversary set A in the experiment above as:

$$Adv_A^{Opriv}(\mathcal{RCD}, f, \kappa) = |Pr[\mathbf{Exp}_A^{Opriv}[\mathcal{RCD}, f, \kappa] = 1 - \frac{1}{2}|$$

A refereed delegation of computation scheme is output private if for the legal encoded output set σ, and for any adversary set A whose members run in probabilistic polynomial time,

$$Adv_A^{Opriv}(\mathcal{RCD}, f, \kappa) \leq neg(\kappa)$$

where $neg()$ is a negligible function of its input.

4 Construction

Main Idea. In the following sections, we only discuss the case where there are two pairs of servers (W_{11}, W_{12}) and (W_{21}, W_{22}), of which one pair of servers is honest. In the following subsections, we will explain how to extend it to N server model. Suppose the user has two sequences λ and μ over some finite alphabet $\Sigma = \{0, \cdots, \sigma - 1\}$ and he wants to delegate the sequence comparison of λ and μ to the cloud servers. We assume each pair of cloud servers cooperatively execute the protocol, but they do not collude with each other. The reason why we use a pair of cloud servers as a server unit will be explained in the following sections.

As stated in Figure 1., the user starts by splitting the sequence λ into λ' and λ'' such that λ' and λ'' are over the same alphabet $\Sigma = \{0, 1, \cdots, \sigma - 1\}$ and $\lambda_i = \lambda_i' + \lambda_i''$ for all $1 \leq i \leq n$. To split λ, the user can first generate a random sequence λ' over the alphabet Σ of length n, and then set $\lambda_i'' = \lambda_i - \lambda_i' \bmod \sigma$, for all $1 \leq i \leq n$. Similarly, the user splits μ into μ' and μ'' such that $\mu_j = \mu_j' + \mu_j''$, for all $1 \leq j \leq m$. Then, λ', μ' are sent to W_{11} and λ'', μ'' are sent to W_{12}. The servers W_{11}, W_{12} collaboratively compute the the edit distance $M(n, m)$ of λ and μ in an additively split fashion. That is, W_{11} and W_{12} each maintain a matrix $M^{(1)'}$ and $M^{(1)''}$ such that $M(i, j) = M^{(1)'}(i, j) + M^{(1)''}(i, j)$ for $1 \leq i \leq n$, $1 \leq j \leq m$. In addition, W_{11} and W_{12} each construct the Merkle Hash Tree for all the values of the matrixes $M^{(1)'}$ and $M^{(1)''}$, and return the result $M^{(1)'}(n, m)$, $M^{(1)''}(n, m)$ together with the value $MH_{root}(M^{(1')}(n, m))$, $MH_{root}(M^{(1'')}(n, m))$, which is the root value of the Merkle Hash Tree for their respective result. The second pair of servers (W_{21}, W_{22}) do the same as described above. Thereafter, the user runs the consistency proof with $(W_{11}, W_{12}), (W_{21}, W_{22})$, and outputs the correct result if at least one pair of servers are honest.

As described in Section 2, a refereed computation delegation of sequence comparison scheme consists of four algorithms (KeyGen, $\mathsf{ProbGen}$, $\mathsf{Compute}$, Verify), which will be formally defined below. The framework of our scheme is presented in Figure. 1.

- $(pk, sk) \leftarrow \mathsf{KeyGen}(\mathrm{f}, \kappa)$: Based on the security parameter κ, the randomized key generation algorithm generates the additively homomorphic encryption [**?**, 33, 34]key pair (pk, sk) for every cloud server and randomly selects a hash function H.
- $(\sigma_x, \tau_x) \leftarrow \mathsf{ProbGen}(x)$: On private input $x = (\lambda, \mu)$, the user splits the sequence λ into λ' and λ'', such that $\lambda = \lambda' + \lambda'' \bmod \sigma$, and similarly splits

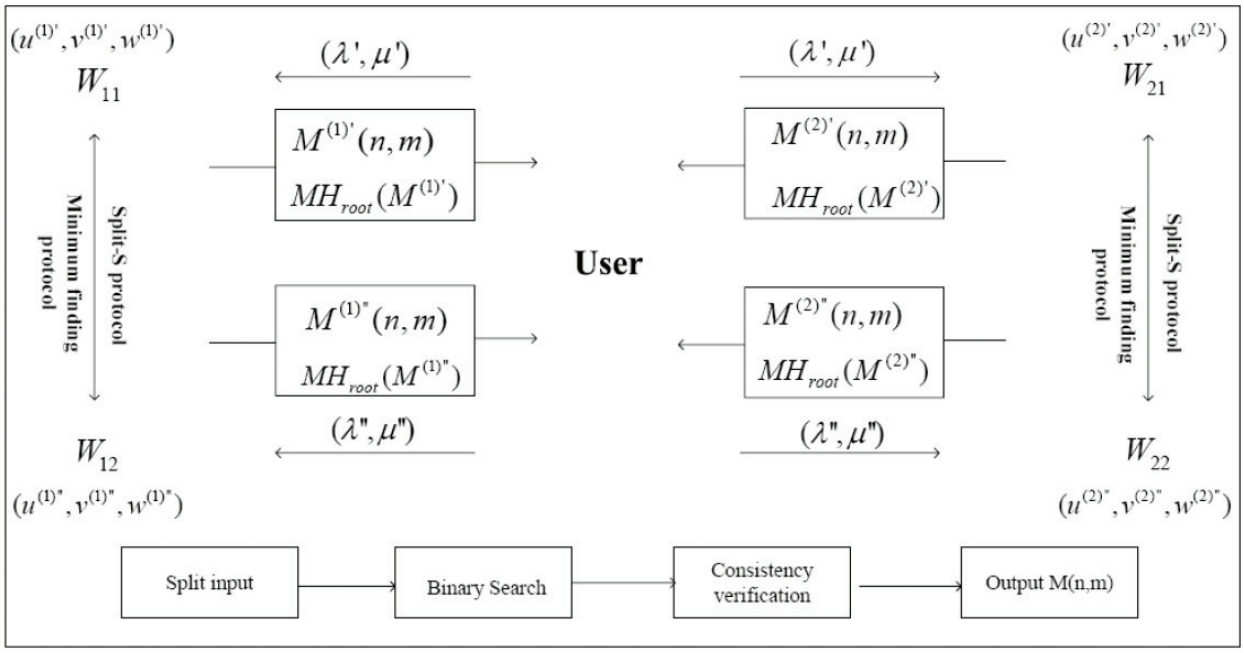

Figure 1: Framework of refereed computation delegation of private sequence comparison

μ into μ' and μ'' such that $\mu = \mu' + \mu''$ mod σ. Meanwhile, U generates two vectors of random numbers, $a = (a_1, \cdots, a_n)$ and $b = (b_1, \cdots, b_m)$. Then U computes two vectors $c = (c_1, \cdots, c_n)$ and $d = (d_1, \cdots, d_m)$ where $c_i = \sum_{k=1}^{i} D(\lambda_k) - a_i$, for $1 \le i \le n$, and $d_j = \sum_{k=1}^{j} I(\mu_k) - b_j$, for $1 \le j \le m$. The outputs of the algorithm is set to be $(\sigma_x, \tau_x)=((\lambda', \lambda'', \mu', \mu'', a, b, c, d), (\lambda, \mu))$. Then U sends λ', μ', b, c to W_{k1}, and sends λ'', μ'', a, d, to W_{k2}, $k = 1, 2$.

- $\sigma_y \leftarrow \mathsf{Compute}(\sigma_x)$: As described in Figure 2. (W_{k1}, W_{k2}), $k = 1, 2$, collaboratively compute $M(i,j)$ for all $1 \le i \le n$, $1 \le j \le m$. Note that W_{k1} owns $M^{(k)}(i,j)'$ and W_{k2} owns $M^{(k)}(i,j)''$, such that $M(i,j) = M^{(k)}(i,j)' + M^{(k)}(i,j)''$.

  The protocol specifications are described as follows:

  1) W_{k1} and W_{k2} first initialize their respective matrix, and then cooperatively compute $(u^{(k)'}, v^{(k)'}, w^{(k)'})$ and $(u^{(k)''}, v^{(k)''}, w^{(k)''})$.
  2) After the minimum finding protocol [3], W_{k1} gets $M^{(k)'}(i,j)$ and W_{k2} gets $M^{(k)''}(i,j)$.
  3) Then W_{k1} and W_{k2} each construct the Merkel Hash Tree for the result matrices $M^{(k)'}$ and $M^{(k)''}$ respectively and return the result and root value to the user. Thus, the output σ_y=$(M^{(k)'}(n,m), MH_{root}(M^{(k)'}), M^{(k)''}(n,m), MH_{root}(M^{(k)''}))$, $k = 1, 2$.

  Note that γ' and γ'' are computed through the following split-S protocol.

  1) W_{k1} generates a $\sigma \times \sigma$ table $\hat{S}$ with $\hat{S}(r,l)$ equals to $E_{pk_{k1}}(S(r + \lambda'_i mod\ \sigma, l))$ and sends it to W_{k2}.
  2) For $1 \le j \le m$, W_{k2} extracts the λ''_i-th row of $\hat{S}$ as a vector v, $v_l = E_{pk_{k1}}(S(\lambda''_i + \lambda'_i mod\ \sigma, l)) = E_{pk_{k1}}(S(\lambda_i, l))$. Then W_{k2} circularly left-shift the vector v μ''_j positions and updates v with a random number γ'', as a result $v_l = E_{pk_{k1}}(S(\lambda_i, \mu''_j + l) * E_{pk_{k1}}(-\gamma'')) = E_{pk_{k1}}(S(\lambda_i, \mu''_j + l) - \gamma'')$.
  3) W_{k1} uses 1-out-of-m oblivious transfer protocol [36]to get the μ'_j-th item of v and decrypts it as $\gamma' = S(\lambda_i, \mu''_j + \mu'_j) - \gamma'' = S(\lambda_i, \mu_j) - \gamma''$.

- $y \cup \perp \leftarrow \mathsf{Verify}_{sk}(\sigma_y)$: U first verifies whether $M^{(1)}(n,m)' + M^{(1)}(n,m)'' = M^{(2)}(n,m)' + M^{(2)}(n,m)''$ or not. In case the equation holds, the answer is correct. Else, the user continues to a binary search as described as follows.

Remark. Assume that the split edit distance matrix is reordered in row-major principle as a vector $\hat{m}$ when the cloud servers construct the Merkle Hash Tree for them. We use variable n_g to denote the good positions which means that $\hat{m}^{(1)'}(n_g)+\hat{m}^{(1)''}(n_g)=\hat{m}^{(2)'}(n_g)+\hat{m}^{(2)''}(n_g)$, and use variable n_b to denote the bad positions. When the binary search ends, U ask W_{11} and W_{12} for the consistency proof at the positions n_g and n_b. If the verification process returns true, U outputs $M(n,m) = M^{(1)}(n,m)' + M^{(1)}(n,m)''$. Otherwise, he outputs $M(n,m) = M^{(2)}(n,m)'+M^{(2)}(n,m)''$. The specifications are presented as follows:

1) U initializes position variables as $n_g = 1$, $n_b = mn$ and asks W_{k1} and W_{k2}, $k = 1, 2$, for the mid-th value

For $k = 1, 2$

Initialization:

W_{k1} sets $M^{(k)'}(0, j) = b_j$, for $1 \le j \le m$, and sets $M^{(k)'}(i, 0) = c_i$, for $1 \le i \le n$.

W_{k2} sets $M^{(k)''}(0, j) = d_j$, for $1 \le j \le m$, and sets $M^{(k)''}(i, 0) = a_i$, or $1 \le i \le n$.

For $1 \le i \le n$, $1 \le j \le m$

(1). W_{k1} computes $u^{(k)'} = M^{(k)'}(i-1, j) + M^{(k)'}(i, 0) - M^{(k)'}(i-1, 0)$
$= M^{(k)'}(i-1, j) + D(\lambda_i) - a_i + a_{i-1}$.

W_{k2} computes $u^{(k)''} = M^{(k)''}(i-1, j) + M^{(k)''}(i, 0) - M^{(k)''}(i-1, 0)$
$= M^{(k)''}(i-1, j) + a_i - a_{i-1}$.

$u^{(k)'} + u^{(k)''} = M^{(k)'}(i-1, j) + M^{(k)''}(i-1, j) = M(i-1, j) + D(\lambda_i)$.

(2). W_{k1} computes $v^{(k)'} = M^{(k)'}(i, j-1) + M^{(k)'}(0, j) - M^{(k)'}(0, j-1)$
$= M^{(k)'}(i, j-1) + b_j - b_{j-1}$.

W_{k2} computes $v^{(k)''} = M^{(k)''}(i, j-1) + M^{(k)''}(0, j) - M^{(k)''}(0, j-1)$
$= M^{(k)''}(i, j-1) - b_j + b_{j-1} + I(\mu_j)$.

$v^{(k)'} + v^{(k)''} = M^{(k)'}(i, j-1) + M^{(k)''}(i, j-1) = M(i, j-1) + I(\mu_j)$.

(3). W_{k1} sets $w^{(k)'} = M^{(k)'}(i-1, j-1) + \gamma'$.

W_{k2} sets $w^{(k)''} = M^{(k)''}(i-1, j-1) + \gamma''$.

$w^{(k)'} + w^{(k)''} = M(i-1, j-1) + S(\lambda_i, \mu_j)$

(4). After the implementation of minimum finding protocol, W_{k1} and W_{k2} get $M^{(k)'}(i, j)$, $M^{(k)''(i,j)}$ respectively, such that

$$M^{(k)'}(i, j) + M^{(k)''}(i, j) = min \begin{pmatrix} M(i-1, j-1) + S(\lambda_i, \mu_j) \\ M(i-1, j) + D(\lambda_i) \\ M(i, j-1) + I(\mu_i) \end{pmatrix}$$

Figure 2: Specifications for the algorithm Compute

of the result vector and the consistency proof of the values, where $mid = (n_g + n_b)/2$. If $\hat{m}^{(1)'}(mid) + \hat{m}^{(1)''}(mid) = \hat{m}^{(2)'}(mid) + \hat{m}^{(2)''}(mid)$, he sets $n_g = mid$, otherwise, he sets $n_b = mid$. U continues the binary search in that way till he gets $n_b = n_g + 1$.

2) W_{11} returns the consistency proof

$$p_{1g} = MH_{poof}(\hat{m}^{(1)'}, n_g), p_{1b} = MH_{poof}(\hat{m}^{(1)'}, n_b),$$

W_{12} returns

$$p_{2g} = MH_{poof}(\hat{m}^{(1)''}, n_g), p_{2b} = MH_{poof}(\hat{m}^{(1)''}, n_b).$$

3) U runs

$$VerMHP(MH_{root}(\hat{m}^{(1)'}), n_g, \hat{m}^{(1)'}(n_g), p_{1g})$$
$$VerMHP(MH_{root}(\hat{m}^{(1)'}), n_b, \hat{m}^{(1)'}(n_b), p_{1b})$$
$$VerMHP(MH_{root}(\hat{m}^{(1)''}), n_g, \hat{m}^{(1)''}(n_g), p_{2g})$$
$$VerMHP(MH_{root}(\hat{m}^{(1)''}), n_b, \hat{m}^{(1)''}(n_b), p_{2b})$$

to verify the consistency proof. If either proof is invalid, the cloud servers W_{k1} and W_{k2} are marked as dishonest. Otherwise, U proceeds as follows.

4) Suppose that $\hat{m}^{(1)'}(n_g)$ is equivalent to $M^{(1)'}(\alpha, \beta)$, then $\hat{m}^{(1)'}(n_g - m)$ and $\hat{m}^{(1)'}(n_g - m + 1)$ are equivalent to $M^{(1)'}(\alpha - 1, \beta)$, $M^{(1)'}(\alpha - 1, \beta + 1)$. As described above, U asks W_{11} and W_{12} for consistency proof at the position $n_g - m$ and $n_g - m + 1$, and then computes

$$M(\alpha - 1, \beta) = M^{(1)'}(\alpha - 1, \beta) + M^{(1)''}(\alpha - 1, \beta)$$

$$M(\alpha-1, \beta+1) = M^{(1)'}(\alpha-1, \beta+1)+M^{(1)''}(\alpha-1, \beta+1).$$

Now that U owns $M(\alpha, \beta)$, $M(\alpha - 1, \beta)$, and $M(\alpha - 1, \beta + 1)$, he can compute $M(\alpha, \beta + 1)$ by himself

using the dynamic programming algorithm. Then U verifies whether $M(\alpha,\beta+1) = M^{(1)'}(\alpha,\beta+1) + M^{(1)''}(\alpha,\beta+1)$ holds (Remember that $M^{(1)'}(\alpha,\beta+1) = \hat{m}^{(1)'}(n_b)$, $M^{(1)''}(\alpha,\beta+1) = \hat{m}^{(1)''}(n_b)$. If the equation holds, U outputs the value of $M(n,m) = M^{(1)'}(n,m) + M^{(1)''}(n,m)$. Otherwise, $M(n,m) = M^{(2)'}(n,m) + M^{(2)''}(n,m)$.

Extensions. Here we show a trivial method of how, given refereed computation delegation scheme with two servers, one can construct a refereed delegation of computation with N servers, where we only need to assume that at least one of them if honest. The idea is to execute the refereed delegation of computation with two servers between each pair of servers. By the soundness of the refereed delegation of computation, with high probability there exists an honest server pair W_i that convinces the user in all of his games. The user outputs the claimed result of W_i.

In addition to the trivial method for extending the protocol to N servers, we can extend this specific protocol also in the following way. Our protocol can be executed with all servers, where the user marks the intermediate value as a good value only if all answers for this position match. At the end of the binary search, the user checks if the computation is consecutive for each one of the servers. After the execution of this protocol, at least one pair of malicious servers will be caught lying and will be declared as a cheater. The user continues to the next round with the other servers, again, executes the protocol to find at least one cheater and then excludes him (or them) from the next rounds. The protocol ends when all the remaining servers agree on the output.

5 Efficiency and Security Analysis

5.1 Efficiency Analysis

First of all, our protocol, which works in the multi-server model, is qualitatively more practical than known techniques for computation delegation in single-server setting. As we know, all known protocols rely either on arithmetization and PCP techniques [18], or rely on fully homomorphic encryption [15,16]. Neither approach is currently viable in practice as a result of low efficiency.

Our protocol neither utilizes the complex proof systems nor fully homomorphic encryption, it is very efficient on both the user side and server side. The computation and communication complexity of splitting the sequence λ and μ and sending the result shares is $O(m+n)$, which is also the lower bound of problem generation algorithm. In the verification procedure, the user searches for inconsistencies between the intermediate values of the two servers' computations. Note that the binary search algorithm ends within $O(\log\ mn)$ steps, where $\log\ m$ and $\log\ n$ are the bit lengths of m and n, respectively. And on finding an inconsistency, the user can detect the cheater by performing a single step of the edit distance computation algorithm. The collision-resistant hash functions are used to allow the server to commit to the large intermediate internal values of the computation using small commitments. And in each verification process of these commitments, the user does $O(\log mn)$ hash computations. Therefore, the overall computation complexity of the user is $O(\log^2(mn))$.

On the server side, each pair of servers runs mn steps of the dynamic program and construct the Merkle Hash Tree for the final result. In each step of the dynamic program, the communication complexity between W_{k1} and W_{k2} is $O(\sigma^2) + O(1)$ due to the computation of γ' and γ'' and the minimum finding protocol. In addition, the computation complexity of constructing the Merkle Hash Tree for the result is $O(mn)$. Therefore, the total computation and communication complexity of each server is $O(\sigma^2 mn)$, which means that the complexity of the server is polynomial in the complexity of evaluating f.

5.2 Security Analysis

Theorem 1. *Suppose that the hash function in use is collision resistant, and the probability that a server honestly executes the protocol is p, then our scheme satisfies the security requirement of correctness against malicious servers with probability 1-$(1-p)^N$, where N is the number of server pairs.*

Proof. Review that in our scheme, the servers construct the Merkle Hash Tree for their respective edit distance matrix M and return the root value MH_{root} to the user. MH_{root} can be viewed as a commitment of M. During the following verification process, the malicious server is unable to change the committed intermediate value of M. Otherwise, if a malicious server tampered the intermediate value but successfully pass the correctness verification, which means that the server generated a fake consistency proof that has the same root value of the Merkle Hash Tree as in the committed edit distance matrix M. Thus, he got a collision in some node along the path to the tampered intermediate value, which contradicts our assumption about the collusion resistant hash function.

In the experiment $\mathbf{Exp}_A^C[\mathcal{RCD}, f, \kappa]$, the ProbGen() oracle and Verify() oracle do not leak useful information for the adversary, as the splitted input sequence λ', μ' are uniformly distributed over the the alphabet set Σ. And we assume that the minimum finding protocol is secure in the two party computation model against malicious adversary, so that the output does not leak any useful information either. In the challenge phrase, after the binary search, U asks W_{11} and W_{12} for the consistency proof for the position n_g and n_b of the result vectors $\hat{m}^{(1)'}$ and $\hat{m}^{(1)''}$. If all the proofs are verified to be valid, U simulates one step of the edit distance computation at the position n_g. The verification equation $M(\alpha,\beta+1) = M^{(1)'}(\alpha,\beta+1) + M^{(1)''}(\alpha,\beta+1)$ holds with probability 1 if the first server is honest and 0 otherwise. For N pairs of servers, the probability that at least one pair of servers are honest is 1-$(1-p)^N$, the probability

is also the probability that the user can get the correct result in our scheme. □

Theorem 2. *Suppose that any pair of servers do not collude with each other, the additively homomorphic encryption scheme is semantically secure, 1-out-of-n OT is receiver secure, and the minimum finding protocol is secure in two-party computation model against malicious adversaries. Then, our scheme satisfies the security requirement of input an output privacy.*

Proof. Firstly, recall that the original sequence λ and μ are splitted into λ',λ'' and μ', μ'', such that $\lambda_i = \lambda'_i + \lambda''_i \mod \sigma$, $\mu_j = \mu'_j + \mu''_j \mod \sigma$. λ'_i, λ''_i, μ'_j and μ''_j are uniformly distributed over the alphabet set $\sum$. Therefore, in the experiment $\mathbf{Exp}_A^{IPriv}[\mathcal{RCD}, f, \lambda]$, the oracle reply of $\mathsf{PubProbGen}(\cdot)$ does not leak any information about the original sequence to the malicious server. Secondly, we assume that the additively homomorphic encryption scheme is semantically secure, which assures that the server W_{k1} does not leak any information about λ' to W_{k2} in the interactive algorithm $\sigma_y \leftarrow \mathsf{Compute}(\sigma_x)$. The OT scheme used in our scheme is receiver secure, so nothing about λ'' is leaked to W_{k1} either. In summarization, our scheme satisfies the security requirement of input privacy. For output privacy, the secure minimum finding protocol used in our protocol guarantees that W_{k1} obtains $M^{(k)'}(i,j)$ but nothing about $M^{(k)''}(i,j)$, meanwhile W_{k2} gets $M^{(k)''}(i,j)$ but nothing about $M^{(k)'}(i,j)$, such that $M(i,j) = M^{(k)'}(i,j) + M^{(k)''}(i,j)$. Therefore, in the challenge phrase, if any adversary is able to corrupt the output privacy our scheme, he can successfully attack the underlying minimum finding protocol with the same probability, which contradicts our assumption. □

6 Conclusion

In this paper, we propose the refereed computation delegation of sequence comparison for the first time. Compared with previous computation delegation schemes, our scheme is qualitatively more practical. Our scheme works in the multi-server model, and the user can get the correct result of the computation as long as there is at least one honest server. For the sequences of length m and n, respectively, our scheme reduce the user computation complexity from $O(mn)$ to $O(\log^2(mn))$. The security analysis shows that our scheme satisfies the security requirements of I/O privacy and correctness.

Acknowledgements

This work is supported by the National Natural Science Foundation of China (No. 61379154, 61100224 and U1135001), the Specialized Research Fund for the Doctoral Program of Higher Education, and Guangdong Natural Science Foundation (No.S2013010013671)

References

[1] B. Applebaum, Y. Ishai, and E. Kushilevitz, "From secrecy to soundness: Efficient verification via secure computation," in *Automata, languages and Programming*, LNCS 6198, pp. 152–163, 2010.

[2] M. J. Atallah and K. B. Frikken, "Securely outsourcing linear algebra computations," in *Proceedings of the 5th ACM Symposium on Information, Computer and Communications Security (ASIACCS'10)*, pp. 48–59, Beijing, China, Apr. 2010.

[3] M. J. Atallah, F. Kerschbaum, and W. Du, "Secure and private sequence comparisons," in *Proceedings of the 2003 ACM Workshop on Privacy in the Electronic Society (WPES'03)*, pp. 39–44, Washington, DC, USA, Oct. 2003.

[4] M. J. Atallah and J. Li, "Secure outsourcing of sequence comparisons," *International Journal of Information Security*, vol. 4, pp. 277–287, 2005.

[5] M. J. Atallah, K. N. Pantazopoulos, J. R. Rice, and E. E. Spafford, "Secure outsourcing of scientific computations," *Advance in Computers*, vol. 54, pp. 215–272, 2002.

[6] L. Babai, "Trading group theory for randomness," in *Proceedings of the Seventeenth Annual ACM Symposium on Theory of Computing (STOC'85)*, pp. 421–429, Rhode Island, USA, May 1985.

[7] S. Benabbas, R. Gennaro, and Y. Vahlis, "Verifiable delegation of computation over large datasets," in *Proceedings of the Crypto'11*, pp. 111–131, Santa Barbara, CA, USA, Aug. 2011.

[8] D. Benjamin and M. J. Atallah, "Private and Cheating-free outsourcing of algebraic computations," in *Proceedings of Privacy, Security and Trust (PST'08)*, pp. 240–245, Fredericton, NB, Oct. 2008.

[9] M. Blanton and M. Aliasgari, "Secure outsourcing of DNA searching via finite automata," in *Data and Applications Security and Privacy*, LNCS 6166, pp. 49–64, 2010.

[10] M. Blanton, M. J. Atallah, K. B. Frikken, and Q. Malluhi, "Secure and efficient outsourcing of sequence comparisions," in *Proceedings of the 17th European Symposium on Research in Computer Security (ESORICS'12)*, pp. 505–522, Pisa, Italy, Sep. 2012.

[11] D. Boneh, G. D. Crescenzo, R. Ostrovsky, and G. Persiano, "Public key encryption with keyword search," in *Proceedings of Eurocrypt'04*, pp. 506–522, Interlaken, Switzerland, May 2004.

[12] R. Canetti, B. Riva, and G. N. Rothblum, "Practical delegation of computation using multiple servers," in *Proceedings of the 18th ACM Conference on Computer and Communications Security(ACM CCS'11)*, pp. 445–454, Chicago, USA, Oct. 2011.

[13] P. S. Chu, C. C. Lee, P. S. Chu, P. S. Chung, and M. S. Hwang, "A survey on attribute-based encryption schemes of access control in cloud environments," *Internation Journal of Network Security*, vol. 15, no. 4, pp. 231–240, 2013.

[14] K. M. Chung, Y. Kalai, and S. Vadhan, "Efficient private matching and set intersection," in *Proceedings of the Eurocrypt'04*, pp. 1–19, Interlaken, Switzerland, May 2004.

[15] K. M. Chung, Y. Kalai, and S. Vadhan, "Improved delegation of computation using fully homomorphic encryption," in *Proceedings of the Crypto'10*, pp. 483–501, Santa Barbara, CA, USA, Aug. 2010.

[16] R. Gennaro, C. Gentry, and B. Parno, "Non-interactive verifiable computing: Outsourcing computation to untrusted workers," in *Proceedings of the Crypto'10*, pp. 465–482, Barbara, CA, USA, Aug. 2010.

[17] C. Gentry, "Fully homomorphic encryption using ideal lattices," in *Proceedings of the Forty-First Annual ACM Symposium on Theory of Computing (STOC'09)*, pp. 169–178, Bethesda, Maryland, USA, May, June 2009.

[18] S. Goldwasser, Y. T. Kalai, and G. N. Rothblum, "Delegating computation: Interactive proofs for muggles," in *Proceedings of the Fortieth Annual ACM Symposium on Theory of Computing (STOC'08)*, pp. 113–122, Victoria, British Columbia, Canada, May 2008.

[19] S. Goldwasser, S. Micali, and C. Rackoff, "The knowledge complexity of interactive proof systems," *SIAM Journal on Computing*, vol. 18, no. 1, pp. 186–208, 1989.

[20] M. Green, S. Honhenberger, and B. Waters, "Outsourcing the decryption of ABE ciphertexts," in *Proceedings of the USENIX Security Symposium*, pp. 34–49, San Francisco, CA, USA, Aug. 2011.

[21] C. Hazay and Y. Lindell, "Efficient protocols for set intersection and pattern matching with security against malicious and covert adversaries," in *Proceedings of the Fifth Theory of Cryptography Conference (TCC'08)*, pp. 155–175, New York, USA, Mar. 2008.

[22] C. Hazay and T. Toft, "Computationally secure pattern matching in the presence of malicious adversaries," in *Proceedings of the Asiacrypt'10*, pp. 195–212, Singapore, Dec. 2010.

[23] M. Jakobsson and S. Wetzel, "Secure Server-Aided signature generation," in *Proceedings of 4th International Workshop on Practice and Theory in Public Key Cryptosystems (PKC'01)*, pp. 383–401, Cheju Island, Korea, Feb. 2001.

[24] A. Jarrous and B. Pinkas, "Secure hamming distance based computation and Its applications," in *Proceedings of 7th International Conference (ACNS'09)*, pp. 107–124, Paris-Rocquencourt, France, Jun. 2009.

[25] S. Kawaumura and A. Shimbo, "Fast server-aided secret computation protocols for modular exponentiation," *IEEE Journal on Selected Areas in Communications*, vol. 11, no. 5, pp. 778–784, 2002.

[26] J. Kilian, "A note on efficient Zero-Knowledge proofs and arguments (extended abstract)," in *Proceedings of the Twenty-Fourth Annual ACM Symposium on Theory of Computing*, pp. 723–732, Victoria, British Columbia, Canada, May 1992.

[27] J. Kilian, "Improved efficient arguments," in *Proceedings of the 15th Annual International Cryptology Conference (Crypto'95)*, pp. 311–324, Santa Barbara, California, Aug. 1995.

[28] C. C. Lee, S. T. Hsu, and M. S. Hwang, "A study of conjunctive keyword searchable schemes," *Internation Journal of Network Security*, vol. 15, no. 5, pp. 321–330, 2013.

[29] J. Li, X. Chen, J. Li, C. Jia, J. Ma, and W. Lou, "Fine-grained access control based on outsourced attribute-based encryption," in *Proceedings of the 18th European Symposium on Research in Computer Security*, pp. 592–609, Egham, UK, Sep. 2013.

[30] J. Li, X. Huang, J. Li, X. Chen, and X. Yang, "Securely outsourcing attribute-based encryption with checkability," *IEEE Transactions on Parallel and Distributed Systems*, vol. 25, no. 8, pp. 2201–2210, 2013.

[31] R. C. Merkle, "A digital signature based on a conventional encryption function," in *Proceedings of CRYPTO'87*, pp. 369–378, Santa Barbara, California, USA, Aug. 1987.

[32] S. Micali, "Computationally sound proofs," *SIAM Journal on Computing*, vol. 30, no. 4, pp. 1253–1298, 2000.

[33] T. Okamoto and S. Uchiyama, "A new Public-Key cryptosystem as secure as factoring," in *Proceedings of the International Conference on the Theory and Application of Cryptographic Techniques (Eurcrypt'98)*, pp. 308–318, Espoo, Finland, May, June 1998.

[34] P. Paillier, "Public-Key cryptosystems based on composite degree residuosity classes," in *Proceedings of the International Conference on the Theory and Application of Cryptographic Techniques (Eurocrypt'99)*, pp. 223–238, Prague, Czech Republic, May 1999.

[35] B. Parno, M. Raykova, and V. Vaikuntanathan, "How to delegate and verify in public: Verifiable computation from attribute-based encryption," in *Proceedings of the 9th International Conference on Theory of Cryptography (TCC'12)*, pp. 422–439, Taormina, Italy, Mar. 2012.

[36] M. O. Rabin, *How to Exchange Secrets by Oblivious Transfer*, Technical Report TR-81, Mar. 1981.

[37] S. Shankar, *Amazon Elastic Compute Cloud*, Technical Report CS 267, Sept. 2009.

[38] J. R. Troncoso-Pastoriza, S. Katzenbeisser, and M. Celik, "Privacy preserving error resilient DNA searching through oblivious automata," in *Proceedings of the 14th ACM Conference on Computer and Communications Security*, pp. 519–528, Alexandria, VA, USA, Oct. 2007.

[39] B. D. Vergnaud, "Efficient and secure generalized pattern matching via fast fourier transformn," in *Proceedings of the 4th International Conference on Cryptology in Africa (Africarypt'11)*, pp. 41–58, Dakar, Senegal, Jule 2011.

[40] R. A. Wagner and M. J. Fischer, "The String-to-String correction problem," *Journal of the ACM*, vol. 21, no. 1, pp. 168–173, 1974.

Xu Ma is a Ph.D. candidate of School of Information Science and Technology, Sun Yat-sen university, China. His research mainly focuses on cryptography and its applications, especially on secure outsourcing computation.

Jin Li received his B.S. (2002) and M.S. (2004) from Southwest University and Sun Yat-sen University, both in Mathematics. He got his Ph.D degree in information security from Sun Yat-sen University at 2007. Currently, he works at Guangzhou University. His research interests include Applied Cryptography and Security in Cloud Computing (secure outsourcing computation and cloud storage).

Fangguo Zhang received his PhD from the School of Communication Engineering, Xidian University in 2001. He is currently a Professor at the School of Information Science and Technology of Sun Yat-sen University, China. He is the co-director of Guangdong Key Laboratory of Information Security Technology. His research mainly focuses on cryptography and its applications. Specific interests are elliptic curve cryptography, secure multiparty computation, anonymity and privacy.

IDuFG: Introducing an Intrusion Detection using Hybrid Fuzzy Genetic Approach

Ghazaleh Javadzadeh[1], Reza Azmi[2]
(Corresponding author: Ghazaleh Javadzadeh)
Sharif University of Technology, International Campus, Kish Island, Iran[1]
(javadzadeh@gjdomain.com)
Dept. of Computer Engineering, Alzahra University, Tehran, Iran[2]
(azmi@alzahra.ac.ir)
(Received Apr. 8, 2013; revised and accepted Feb. 23 & June 3, 2014)

Abstract

In this paper, we propose a hybrid approach for designing Intrusion Detection Systems. This approach is based on a Fuzzy Genetic Machine Learning Algorithm to generate fuzzy rules. The rules are able to solve the classification problem in designing an anomaly IDS. The proposed approach supports multiple attack classification. It means that, it is able to detect five classes consist of Denial of Service, Remote to Local, User to Root, Probing and normal classes. We present a two-layer optimization approach based on Pittsburgh style and then combine it with Michigan style. To improve the performance of the proposed system, we take advantages of memetic approach and proposed an enhanced version of the system. We test it on NSL KDD data set to be able to compare our works with previous ones. As results show our approach can converge faster to the classification accuracy about 98.2% and 0.5% false alarm.

Keywords: Computational intelligence, fuzzy genetic, intrusion detection, multiple attack classification, soft computing

1 Introduction

Intrusion Detection System (IDS) is a security tool that detects a malicious behavior called attack. An IDS may be a software or a combination of software and hardware [13]. IDS monitors and analyzes the data within a computer or a network to detect and alert the system if an attack happens. It should be mentioned that an IDS do not react against attacks, but it just detect and alert the system or the network administrator. The function of reacting against attacks is undertaken by other security mechanisms. However, IDS plays a critical role in security of networks and systems since detection is the first step in controlling an attack.

We can classify IDS from different aspects, the most common is the source of audit data; accordingly there are two types of IDSs: Host based (HIDS) and Network based (NIDS). HIDS surveys the data of a single host such as the operating system kernel logs, and application program logs;While NIDS gathers and analyzes the data transmit between several hosts in a computer network. Actually a NIDS examines the traffic packets to detect intrusion patterns. Although, each of these systems has its own advantage and disadvantage. It is recommended to use a combination of both for more security. Figure 1 illustrates the Host based and the Network based intrusion detection system.

There are several detection methods that an IDS use to detect intrusions. Two main detection methods are Signature based (misuse) and Anomaly based detection. In the former, the IDS compares the data traffic with

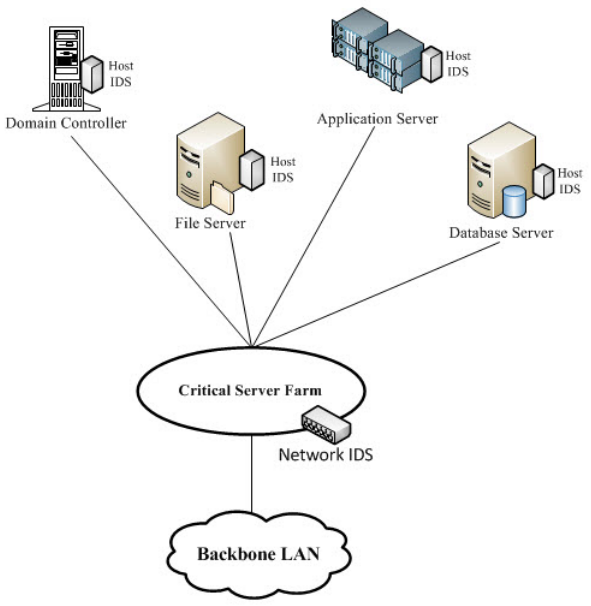

Figure 1: Host based and network based IDS

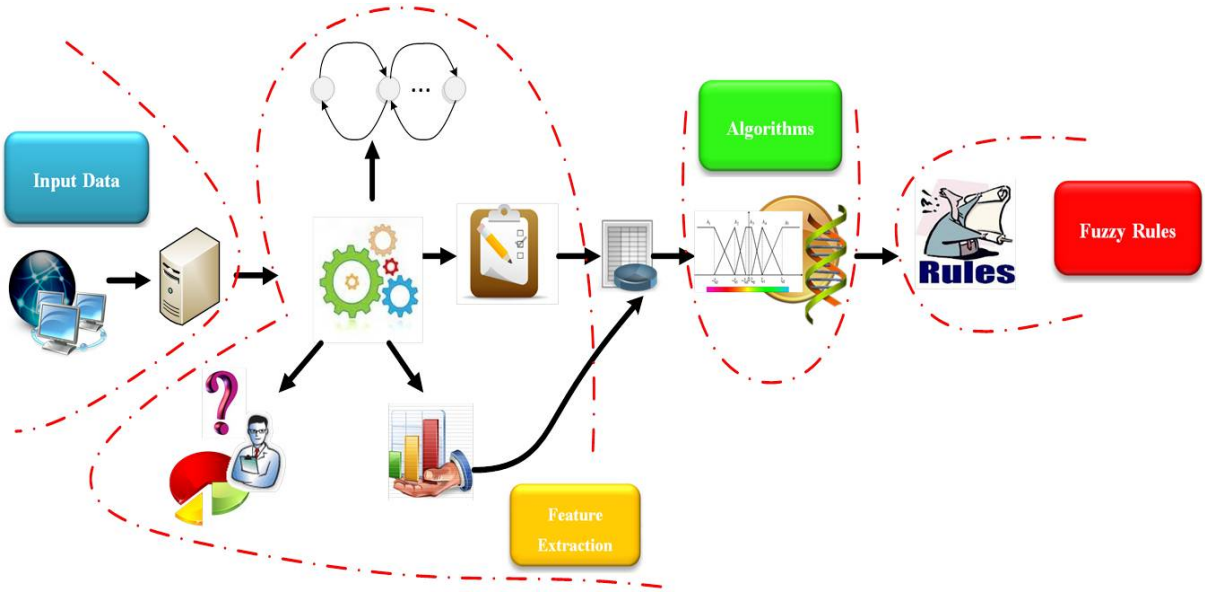

Figure 2: Host based intrusion detection system

some predefined patterns of attacks, and if they match, the traffic will be detected as an attack. In this method the rate of false alerts reduces, however attacks with new patterns cannot be detected.

Anomaly detection method detects data traffic as an attack if it is not matching the normal traffic. In fact, a normal traffic is defined in a system, and IDS captures and compares the traffic with it. If a mismatch happens, it means that an attack happens. In this method more attacks with new patterns can be detected while we may encounter more false alarms too. In this paper we study both types of IDSs but propose an anomaly detection approach for NIDS.

There are two important challenging points when reviewing the IDS literature, the dataset and the algorithm. Selecting a proper algorithm to train the IDS and using suitable dataset to test it, affect the detection ability. Many researchers have done efforts in this domain. Numerous methods for detection are invented with the goal of high classification accuracy, beside low false alarm, high computational speed, and low computational complexity. Figure 2 illustrates a host based intrusion detection system.

The first challenging point is selecting a proper algorithm to detect intrusion. In most of the previous works the detection results in a binary state which shows the traffic is an attack or not. Besides we find out that an action is an attack, it is important to know the type of that attack, because different type of attacks threat different security aspects. Some security attacks, disclose the confidentiality of our assets while some others alter integrity or the impossible availability of them. When we know the attack type, we can pick the most proper security mechanism to encounter with it.

In this paper, we propose an algorithm with different structures for multiple classification of the attack traffics, means we present a multiple classification of detection results. For example, it classifies the traffic as the normal user behavior, and other of the intrusion types such as Probing, Denial of Service (DOS), and User to Roots attacks (U2R), and so on. Actually the type of attack will be determined in this kind of classification. The continual changing nature of attacks requires a flexible defensive system that is capable of analyzing the enormous amount of traffic in a manner which is less structured. The ability of performing multiple attack classifications is one of the most important features of our proposed approach. The difference between multiple classification and binary classification is illustrated in Figure 3.

Studies show that statistical techniques such as Hidden Markov Model, Multivariate Adaptive Regression Splines, Bayesian Classifier and Classification and Regression Tree have been applied to intrusion detection. These statistical approaches usually result in an inflexible detection system that is unable to detect an attack if the sequence of events is different a little bit from the predefined profile. Recently, computational intelligence has been successfully applied for developing IDS.

As mentioned in [25] characteristics of computational intelligence systems, such as adaption, fault tolerance, high computational speed and error resilience in the face of noisy information, fit the requirements of building a good intrusion detection model. According to the classification of their article, in order to design an algorithm for an IDS using computational intelligence we can apply following techniques: artificial neural networks, fuzzy sets, evolutionary computation, artificial immune system, swarm intelligence, and soft computing. Figure 4 represents the mentioned classification of these techniques.

Among these techniques Soft Computing has a particular place. In fact, soft computing is a method in which, combination of other computational intelligence

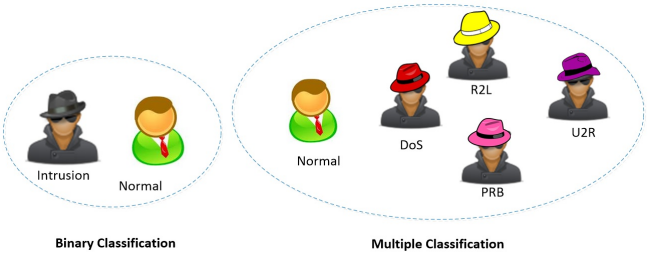

Figure 3: Multiple classifications against binary classification

techniques are used. As a soft computing approach, we have chosen a mixture of evolutionary computation and fuzzy systems for designing an intrusion detection algorithm. There are two major reasons for using fuzzy in IDS. First, there are many numeric attributes in the collected audit data and various derived statistical measures. Second, since fuzzy logic deals with imprecise information and the essence of security includes fuzziness as boundaries between normal and abnormal are not well define.

Genetic Base Machine Learning or GBML is a type of Evolutionary Algorithm (EA); EA is a family of search and optimization approaches that is inspired of the natural principles of evolution [22]. Actually, we proposed an intrusion detection system for known and unknown attacks (anomaly detection) using one of the computational intelligence techniques, as shown in Figure 4.

Among all computational intelligence techniques, we have chosen a mixture of evolutionary computation and fuzzy systems called GBML algorithm [14]. GBML has two main approaches: Pittsburgh and Michigan. Each of them has some advantages and disadvantages which have been mentioned in detail in [22]. Our proposed algorithm is a combination of these two approaches; by this means it would be possible to use the advantage of both algorithms such as the direct optimization of rules, high search ability, and at the same time, shorter computational time. In fuzzy genetic algorithms, each individual can be a rule or rule set depending on the approach. Pittsburgh style handles the rule sets as individuals while rules are considered as individuals in Michigan.

As mentioned above, it is important to use a proper dataset for the test phase. Of course, the dataset contains features of normal and abnormal behavior use for designing and testing intrusion detection systems. There are two way of selecting datasets. The first one is to create your own dataset and the second is using existing datasets such as KDD Cup99 [1] or NSL KDD [2]. Many researchers used KDD CUP99 dataset (KDD). Mahoney and Chani [20] criticized the dataset validity. As they claim information in the dataset does not look like a real traffic in many aspects. Based on their analysis, the IDS created by using KDD with low false alarm may generate high false alarms in real environment.

NSL KDD dataset is an improved version of KDD CUP 99 [2]. This dataset has been used for the evaluation of anomaly based detection methods. The data set was generated by gathering the network logs. It contains two parts, train and test. Patterns of this dataset contain 41 attributes. Figure 5 shows a snapshot of NSL KDD dataset.

The best advantage of using existing datasets is that we can evaluate our work with others. But actually because none of the existing datasets contains all kind of todays attacks it is recommended to create your own dataset with all possible kinds of attacks. Also for different websites and web applications we may encounter different attacks, so it is better to gather the normal and abnormal behaviors of their users for each one separately. In this paper we use the NSL KDD to test our proposed algorithm, because it let us compare our algorithm with previous works.

Briefly, in this paper, we propose an intrusion detection system called IDuFG that is capable of doing multiple attack classifications. The algorithm of this system uses a soft computing technique that is a fuzzy genetic approach. Test and train of the proposed algorithm is been done on NSL KDD dataset.

The rest of this paper is organized as follows: Section 2 is the literature review and also an overview of the related work in terms of intrusion detection systems. In Section 3 and Section 4, the proposed approach and simulation results are explained respectively. Section 5 concludes the paper and presents the future work.

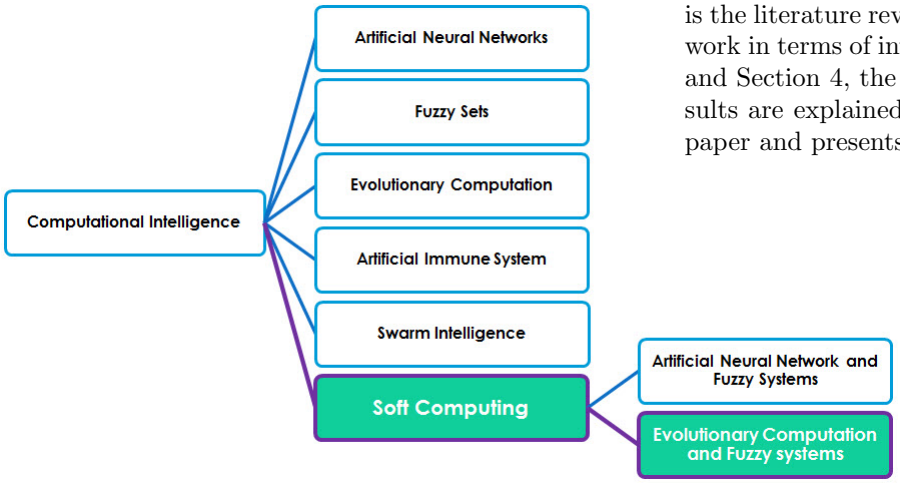

Figure 4: Computational intelligence techniques

0,tcp,ftp_data,SF,491,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,2,2,0.00,0.00,0.00,0.00,1.00,0.00,0.00,150,25,0.17,0.03,0.17,0.00,0.00,0.00,0.05,0.00,normal,20
0,udp,other,SF,146,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,13,1,0.00,0.00,0.00,0.00,0.08,0.15,0.00,255,1,0.00,0.60,0.88,0.00,0.00,0.00,0.00,0.00,normal,15
0,tcp,private,S0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,123,6,1.00,1.00,0.00,0.00,0.05,0.07,0.00,255,26,0.10,0.05,0.00,0.00,1.00,1.00,0.00,0.00,neptune,19
0,tcp,http,SF,232,8153,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,5,5,0.20,0.20,0.00,0.00,1.00,0.00,0.00,30,255,1.00,0.00,0.03,0.04,0.03,0.01,0.00,0.01,normal,21
0,tcp,http,SF,199,420,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,30,32,0.00,0.00,0.00,0.00,1.00,0.00,0.09,255,255,1.00,0.00,0.00,0.00,0.00,0.00,0.00,0.00,normal,21
0,tcp,private,REJ,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,121,19,0.00,0.00,1.00,1.00,0.16,0.06,0.00,255,19,0.07,0.07,0.00,0.00,0.00,0.00,1.00,1.00,neptune,21
0,tcp,private,S0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,166,9,1.00,1.00,0.00,0.00,0.05,0.06,0.00,255,9,0.04,0.05,0.00,0.00,1.00,1.00,0.00,0.00,neptune,21
0,tcp,private,S0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,117,16,1.00,1.00,0.00,0.00,0.14,0.06,0.00,255,15,0.06,0.07,0.00,0.00,1.00,1.00,0.00,0.00,neptune,21
0,tcp,remote_job,S0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,270,23,1.00,1.00,0.00,0.00,0.09,0.05,0.00,255,23,0.09,0.05,0.00,0.00,1.00,1.00,0.00,0.00,neptune,21
0,tcp,private,S0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,133,8,1.00,1.00,0.00,0.00,0.06,0.06,0.00,255,13,0.05,0.06,0.00,0.00,1.00,1.00,0.00,0.00,neptune,21

Figure 5: An NSL KDD dataset snapshot [2]

2 Litrature Review

In our research, we study different approaches that have been used in designing intrusion detection systems. Numerous approaches consist of pure or hybrid, have been developed in recent years. It seems that hybrid algorithms that use a combination of fuzzy logic and other methods are attracting issues in the area of designing intrusion detection systems.

Genetic algorithm includes search algorithms that are inspired from biological evolution as a problem-solving strategy by using some natural mechanism such as generation, mutation, crossover, and selection. Several Genetic Based Machine Learning (GBML) algorithms have been proposed for designing fuzzy rule-based systems. As mentioned in [5] GAs are search algorithms that theoretically and empirically are proven to have a great capability in complex space in search and optimization problems, so will be used in pattern classification problems commonly. Also brilliant characteristics of genetic algorithm, such as automatic optimizing, global researching, and adaptability, is the reason for choosing it.

Several fuzzy GBML algorithms have been proposed for designing fuzzy rule-based systems. Fuzzy GBML algorithms can be also classified into two categories: Michigan approach and Pittsburgh approach. In [22] Nojima et al. developed two GA-based schemes for the design of fuzzy rule-based classification systems. Search ability of fuzzy genetic rule selection and fuzzy GBML algorithms compare to each other.

Cintra and Camargo in [8] named Rule Based definition as one of the most important and difficult tasks in designing fuzzy systems. They introduce a fuzzy rule base generation method using the genetic algorithm. The algorithm support binary classification. This algorithm includes a phase of pre-selection of candidate rules that has been proposed by the authors. The use of a self-adaptive algorithm for the fitness calculation in genetic algorithm is proposed as an improvement of the mentioned method. The advantage of proposed method has been tested by some experimental results.

A parallel genetic local search algorithm is presented in [4]. According to that paper, an important solution to reduce the false alarm rate in detection intrusions is the fuzzy logic. This algorithm produces fuzzy rules for network intrusion detection systems. This system use Michigan approach. As shown in Figure 6, in this algorithm, the total population is divided into subpopulations. Each subpopulation contains the same class fuzzy rules.

An intrusion detection approach that extracts accurate and interpretable fuzzy rules for classification has been proposed in [23]. They use a statistical pattern recognition approach to design an intrusion detection system. These rules are obtained from the network traffic data. They evaluated their approach using KDD CUP and compared the results with some well-known classifiers. In that paper fuzzy rule generation strategy is discussed. The problem of this method is high false alarm ratio.

A genetic fuzzy system is proposed in [6] and the authors compared it with other approaches in intrusion detection systems. As the results show, it is obvious that because false alarm rate in a fuzzy genetic system, the based intrusion detection is lower than other approaches, IDS which develops using genetic fuzzy systems would be more reliable than other approaches.

Also, a new genetic fuzzy logic method for automatic rule generation has been proposed in [26]. This algorithm automatically adjusts the crossover rate and the mutation rate using rules population diversity and evolutionary speed. The simulation results in that paper indicate that the algorithm is practical and effective. The disadvantage of this algorithm is that in cases that the boundaries of samples are not clearly defined, the rules of different

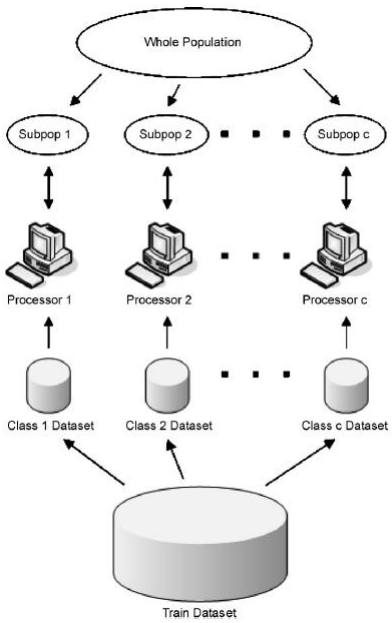

Figure 6: The parallel learning framework [4]

classes overlap with each other.

A hybrid algorithm proposed by H. Ishibuchi in [14] that has the advantages of the two fuzzy GBML algorithms. The basis of the algorithm is Pittsburgh that Michigan-style algorithm is applied to each rule set after the mutation operation to generate new fuzzy rules. Another hybrid algorithm was proposed in [15] for designing fuzzy rule based systems for pattern classification problem. This algorithm is a combination of two Genetic-Based Machine Learning algorithms called Pittsburgh and Michigan. The experimental results show that this algorithm has high search ability that Michigan and Pittsburgh approaches. To obtain a better tradeoff between the accuracy of the system and its complexity, this algorithm should be extended to the multi objective design of fuzzy rule-based classification system. The algorithms have high complexity.

In this paper, since most of the related work and similar algorithms use KDD dataset to test their proposed methods, we also test our proposed algorithms on NSL KDD to be able to compare our work with them. As our studies shows, each of the previous works has its own advantages and disadvantages. None of them covers all of the requirements of a good IDS. As shown in Table 1, the most important challenging issues in designing an IDS are performance, detection rate, computational time, ability to detect new attacks, and false alarm.

Table 1: Comparison of fuzzy genetic systems

	Method	Category	Pros	Cons
[8]	Fuzzy Rules Generation using Genetic algorithms with Self-adaptive Selection	Fuzzy Genetic	Speed up the search process	False alarms
[4]	Parallel Genetic Local Search	Fuzzy Genetic	High performance	Long computational times
[23]	Genetic Fuzzy Rule Mining Approach	Fuzzy Genetic	High detection rate	Long computational time
[6]	Genetic Fuzzy System Based	Fuzzy Genetic	High performance	Unable to detect new attacks
[26]	Automatic fuzzy rule generation using fuzzy genetic algorithm	Fuzzy Genetic	Practical and effective in applications	Long computational time
[15]	Hybridization of Fuzzy GBML approaches	Fuzzy Genetic	High search ability	There is no tradeoff between accuracy and complexity of system

3 Proposed Approaches

We propose a fuzzy based genetic algorithm for designing Intrusion Detection Systems in two Basic and Enhanced approaches. Our basic proposed approach is a combination of two Genetic Based Machine Learning styles called Michigan and Pittsburgh; by this means it would be possible to use the advantage of both styles such as direct optimization of rules, high search ability, and at the same time shorter computational time. In this section, we briefly introduce genetic algorithm, Pittsburgh and Michigan styles, fuzzy rules, and finally our proposed fuzzy-based approaches will be described in the last section.

3.1 Genetic Algorithm

Here, we describe the general process of the genetic algorithm. Genetic algorithms are an effective search method in cases that the solution space is wide and large that lead to find optimized solutions of a problem. This algorithm works with a series of coded variables. So we should code our variable to be able to find the optimized solution for them. In using the genetic algorithms, three following concepts should be determined:

1) Defining the objective function or the cost function
2) Genetic representation
3) Defining and implementation of the genetic operators

The general process of the genetic algorithm is illustrated in Figure 7; in this algorithm, at first an initial population is generated randomly, then the fitness of each individual is evaluated. According to the fitness value of each one two parents are selected. Then offspring will be produced by using the genetic operator such as crossover and mutation. This offspring is an individual of the next generation.

In subsequent iteration the initial population is an improved version rather random. At the end of iterations stopping condition is checked. As studies show there are several stopping criteria such as:

1) A fixed number of iterations
2) A fixed amount of time
3) Run until convergence has occurred

Genetic algorithm is applied on a set of solutions called population. Usually populations consist of 20 to 100 individuals. Most custom method to indicate individuals in genetic algorithm is in a string form. One of the important points that should be considered is that after applying ge-

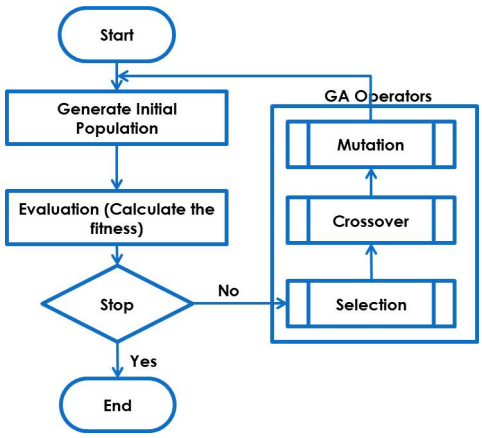

Figure 7: Fuzzy genetic algorithm flowchart

netic operators and producing new results, it is possible to reach the answers that are not satisfying the problem conditions. A simple solution to this problem is using penalty function, by this means fitness of the individuals that not match the conditions will decrease excessively.

Usually, to produce next generations, three fundamental operators are used: Selection, Crossover, and Mutation. The new generation is assessed according to the stopping condition. If the stopping condition is not satisfied this process will repeat to generate a more optimal generation. There are different kinds of operators; according to our studies and their comparisons between several genetic operators, we use tournament selection and uniform crossover, in our proposed approach.

3.2 Fuzzy GBML Styles

Fuzzy GBML algorithms have two main categories: Michigan and Pittsburgh. According to [15], in Michigan-style algorithms, there is a population consisting of a prespecified number of fuzzy rules. A population of individuals generates new populations by the operators such as crossover and mutation. The procedure of Pittsburgh algorithm is the same as Michigan with partial differences.

Each individual in Michigan is a rule and the population is a collection of these rules. As a comparison of these two styles: in Pittsburgh individual is a rule set, fitness define for a rule set, optimization is direct, computation time is long and memory usage is large while in Michigan, the individual is a rule, fitness define for a rule. Also it has indirect optimization, with short computation time and small memory usage [6]. The differences of these two approaches are summarized in Table 2.

Michigan.
The Michigan approach is characterized by the fuzzy rules as its individuals, and the whole population is the solution to the classification problem [12]. Detail description of coding rule, operators and fitness calculation is available in [9]. Here we just point to the general form of a Michigan approach. According to Michigan Style Fuzzy GBML Algorithm mentioned in [15], the approach is as follow (Algorithm 1):

Table 2: Comparison of Pittsburgh and Michigan [16]

Approach	Pittsburgh	Michigan
Individual	A rule set	A rule
Fitness definition	For a rule set	For a rule
Optimization	Direct	Indirect
Elite	Rule sets	Rules
Inheritance of good rules	×	√
Inheritance of good rule sets	√	×
Computation time	Long	Short
Memory storage	Large	Small

Algorithm 1 Michigan style fuzzy GBML

1: Begin
2: Generate N_{rule} fuzzy rules as initial population.
3: Calculate the fitness value of each rule.
4: Generate M rules using genetic operations.
5: Use M new rules and $(N_{pop} - M)$ best rules of current population to produce the next generation.
6: If the stopping condition is not satisfied go to step 2.
7: End

As it is obvious, in all steps of this algorithm, fuzzy rules are considered as individuals. Therefore the fitness value is calculated for each rule and also the operators apply on rules.

Pittsburgh.
In this approach each individual is a complete solution to the problem. A complete solution means a rule set. Detail description of coding rule, operators and fitness calculation is available in [9]. Here we just point to the general form of a Pittsburgh approach. According to Pittsburgh Style Fuzzy GBML Algorithm mentioned in [15], the approach is as follow (Algorithm 2).

Algorithm 2 Pittsburgh style fuzzy GBML

1: Begin
2: Generate N_{pop} rule sets with N_{rule} fuzzy rules as initial population.
3: Calculate the fitness value of each rule set.
4: Generate M rule sets using genetic operations.
5: Use M new rule sets and $(N_{pop} - M)$ best rule sets of current population to produce the next generation.
6: If the stopping condition is not satisfied go to step 2.
7: End

To takes advantage of both approaches; we use a combination of them in our proposed approaches. Michigan yields good rules but not necessarily good rule-sets while Pittsburgh yields good rule sets but not good rules. But

our proposed algorithm enjoys good rule sets with good rules. Also this algorithm has the high search ability of the Michigan, as well as the direct optimization ability of the Pittsburgh.

3.3 Fuzzy Rules

In fuzzy genetic algorithms, each individual can be a rule or rule set depending on the approach. Pittsburgh style handle the rule sets as an individual while rules are considered as individual in Michigan. In this section, the procedure of rule generation is described. This process is one of the important steps in our proposed algorithm. In our proposed algorithms we use fuzzy rules of the following type:

$$R_i = if\ X_1\ is\ A_{i1}\ \&\ ...\ \&\ X_n\ is\ A_{in} \quad then\ class\ C_i\ with\ CF_i\ ,\ i = 1, 2, ..., n. \quad (1)$$

Where R_i is a fuzzy rule and X_i refers to the i^{th} feature of the corresponding pattern, A_{in} is an antecedent fuzzy set with linguistic label, C_i is a consequent class and CF_i is a rule weight and n is the number of rules.

In this stage we started to find the class of each rule [19]. To do so, the compatibility of antecedent fuzzy rules with the random pattern is calculated by equation.

$$\mu_i(X_P) = \mu_i(X_{P1}) \times \ldots \times \mu_{in}(X_{Pn}), P = 1, 2, \ldots, m \quad (2)$$

where $\mu_{in}(.)$ is the membership function of A_{in} that achieve from five linguistic values shown in Figure 8 and m is the number of patterns. The consequence class of each rule is determined according to following equation. Figure 9 shows a sample of a fuzzy rule.

$$\beta_{class\hat{h}_i}(R_i) = max\{\beta_{class_1}(R_i), \ldots, \beta_{classh}(R_i)\} \quad (3)$$

where

$$\beta_{classh}(R_i) = \frac{\sum_{xp \in classh} \mu_i(xp)}{N_{classh}} \quad h = 1, 2, \ldots, n(numberofclasses). \quad (4)$$

Each of the fuzzy rules in the final classification has a certainty grade (class weight), that means the strength of that fuzzy rule and calculated according to:

$$CF_j = \frac{(\beta_{class\hat{h}_i}(R_i) - \overline{\beta})}{\sum_{h=1}^{5} \beta_{classh}(R_i)'} \quad (5)$$

where

$$\overline{\beta} = \frac{\sum_{h \neq h_i} \beta_{classh}(R_i}{n-1}. \quad (6)$$

Certainty grade of each fuzzy rule is a number in [0,1] interval that indicates the accuracy amount of the consequence part of a rule according to the accuracy of the antecedent part of that rule. So rules can be generated accordingly and also the rule sets that are a group of rules.

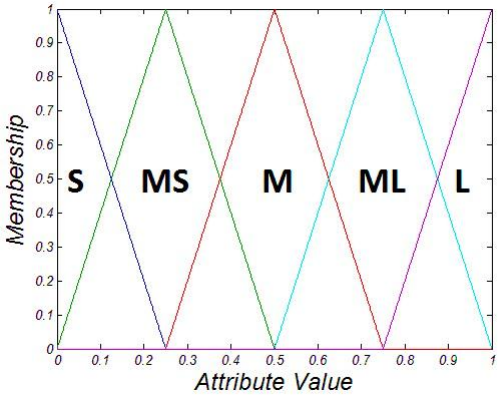

Figure 8: Membership function of five linguistic value [5]

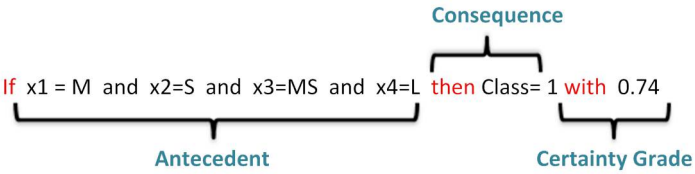

Figure 9: Fuzzy if-then rule

According to the approach that we pick, rules or rule sets are considered as individuals.

After generating N_{pop} rule sets with N_{rule} fuzzy rules as initial population using the above method, we should evaluate the fitness of each individual. Fitness value for each rule is calculated by (N_{pop} and N_{rule} indicate the number of rule sets and the number of rules in a rule set):

$$fitness(R_j) = w_t * T_p - w_f * F_p \quad (7)$$

Where T_p is the number of correctly classified training patterns and F_P is the number of incorrect classified training patterns but the fitness of the rule set is just the number of the correctly classified patterns. w_t and w_f are the weights of correct or incorrect classification, respectively. In fact, each rule is evaluated by classifying the given training patterns.

One of the important points that should be considered is that after applying genetic operators and producing new results, it is possible to reach the answers that are not satisfying the problem conditions. A simple solution to this problem is using penalty function, by this means fitness of the individuals that not match to the conditions will decrease excessively. In the above equation the w_f plays the role of penalty function.

Since any alteration in membership functions leads to vague results, in this paper we do not apply any changes in membership function of input. In the proposed fuzzy GBML algorithm, we use the above membership function for all inputs.

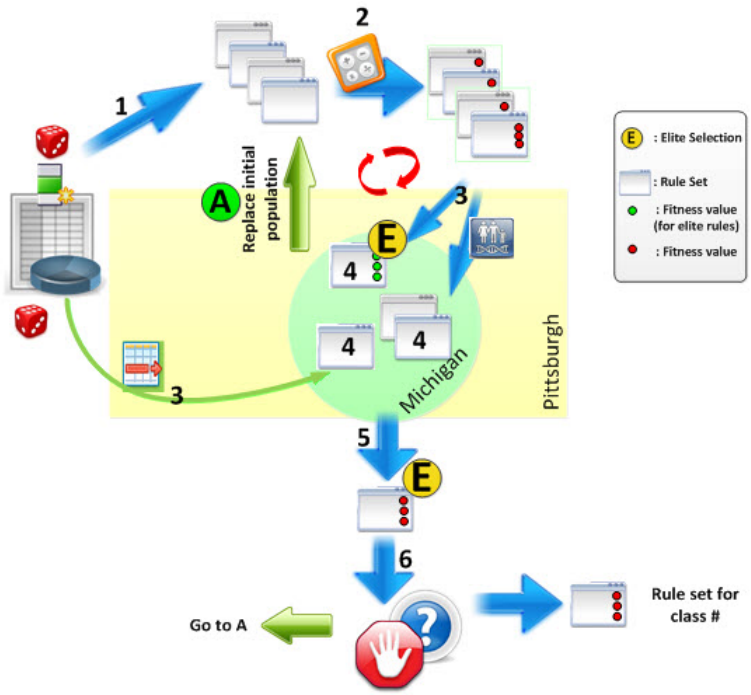

Figure 10: IDuFG algorithm for each class (Numbers 1-6 indicates the algorithm steps)

3.4 The Proposed Approaches

We propose a fuzzy based genetic algorithm for designing Intrusion Detection Systems that is divided in Basic and Enhanced approaches. Our basic proposed algorithm is a combination of these two approaches; by this means it would be possible to use the advantage of both algorithms such as direct optimization of rules, high search ability, and at the same time the shorter computational time. Accordingly in Subsection 3.4.1 we explain the Basic approach and the enhanced approach will be explained in Subsection 3.4.2.

3.4.1 Basic Approach

The goal of a fuzzy GBML algorithm is to find the small number of fuzzy if-then rules with high classification anility. Our proposed algorithm is a combination of two fuzzy GBML algorithms; Pittsburgh is selected as the base of the algorithm and at the end of iterations of Pittsburgh we have a single iteration of Michigan. Initial population is generated according to the random training patterns. N_{pop} rule sets are generated from the random patterns of the training data set as the initial population. Figure 10 illustrates the process of result rule set generation for each class. The algorithm is running for each class separately, and the final result contains all rule sets of all classes. Our proposed algorithm is written in Algorithm 3.

To produce an initial population of fuzzy rule sets, we pick N_{pop} training patterns randomly, and generate combinations of antecedent part of several fuzzy rules according to the selected patterns.

Authors of [7] discussed about how to extract fuzzy rules directly from numerical data for pattern classification. Each rule set is generated according to a training

Algorithm 3 Basic approach

1: Begin
2: Generate N_{pop} rule sets with N_{rule} fuzzy rules.
3: Calculate the fitness value of each rule set in the current population.
4: Generate next generation according to the following portions (experiments show that the following percentage producing best results):

- 80% rule sets of the next generation by the selection, crossover and mutation in the same manner as the Pittsburgh-style algorithm.
- 10% of the next generation is formed by elite rule set selection.
- 5% is elite rule selection among all rule sets (i.e. the best rules of all rule sets).
- 5% is the updated patterns from the training patterns that not choose in the first step.

5: Apply a single iteration of the Michigan style algorithm (i.e., the rule generation and the replacement) to each of the generated rule sets.
6: Choose the best rule set in the new population as the result of this iteration.
7: Return to Step 2 if the pre-specified stopping condition is not satisfied.
8: End

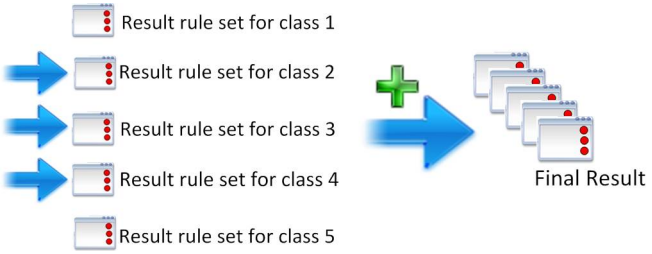

Figure 11: Final result of basic approach

pattern. As mentioned before, an algorithm is running for each class separately; so in all steps the rules which are in the same class of algorithm running for will be accepted.

In Step 3 that we apply genetic algorithms, we use tournament selection for selecting the parent rules or rule sets, by this means we give the chance of contribution in forming the next generation to the rules or rule sets that have not the high fitness value. In this method a subset of individuals is randomly selected, and then these individuals compete with each other according to their fitness . To achieve the offspring we use uniform crossover with probability of Pc and mutation with probability of Pm. Each part of individuals is mutated independent of other parts. It means the mutation of a rule do not influence on the mutation probability of other parts.

After producing the next generation we should check the stopping condition. If the pre-specified stopping condition is not satisfied we should return to the step 2; in our proposed basic approach, the number of iterations is considered as the stopping conditions. This time instead of starting with a random population, use the newly produced generation. The Result of the algorithm running for each class is the best rule set which means the rule set with the highest fitness value. As shown in Figure 11 the final result of the algorithm contains all result rule sets of all classes. The flowchart of the basic approach is illustrated in Figure 12.

3.4.2 Enhanced IduFG

The second approach is an enhanced version of the basic approach. Producing the result for each class in this approach is the same as the previous one, while the formation of the final result has been done in a different way. As Figure 11 indicates, in the basic approach the final result consists of all the rules in rule set of each class, but as Figure 13 shows the final result contains the elite rule of the rule set of each class. Therefore the number of the final rules is equal to the number of a rule set.

Figure 14 illustrates the flowchart of the second approach. The same as the basic approach, the algorithm is running for each class separately and generates a rule set as a result of the algorithm for each class, but against the basic approach, to generate the final result, in this approach, elite rules of these rule sets will be selected to form the final result.

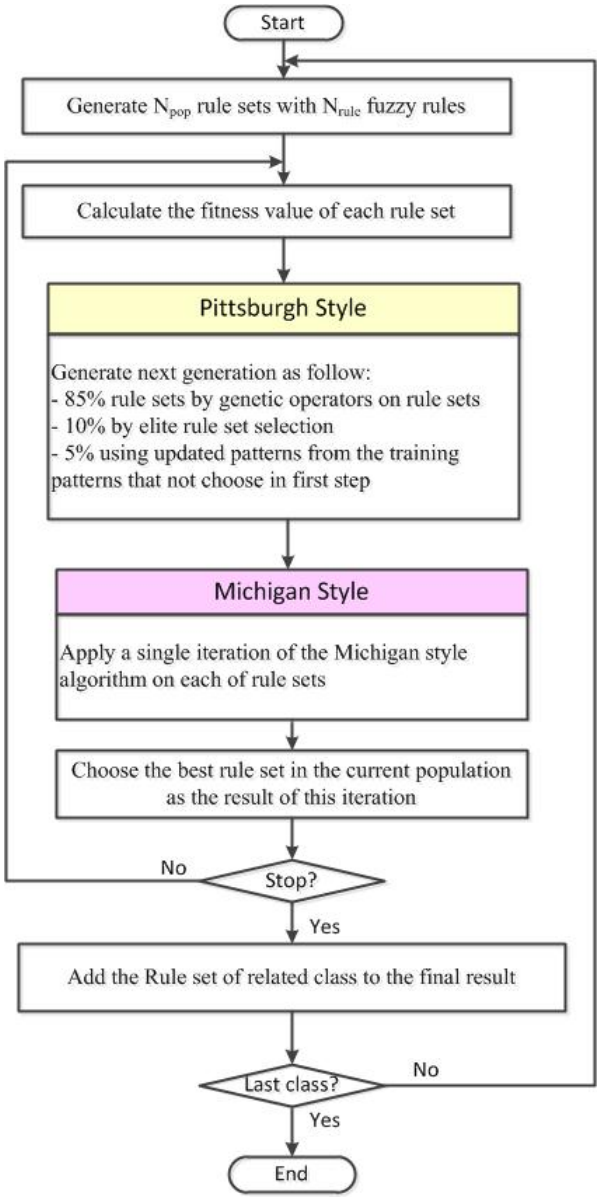

Figure 12: The basic approach flowchart

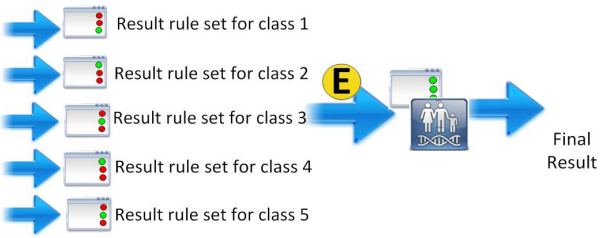

Figure 13: Final result of enhanced approach

4 Simulation Results and Evaluation

In this section, we show the results of the implementation of our proposed algorithm and compare it with one of the best and nearest work proposed in [15]; from now on, we

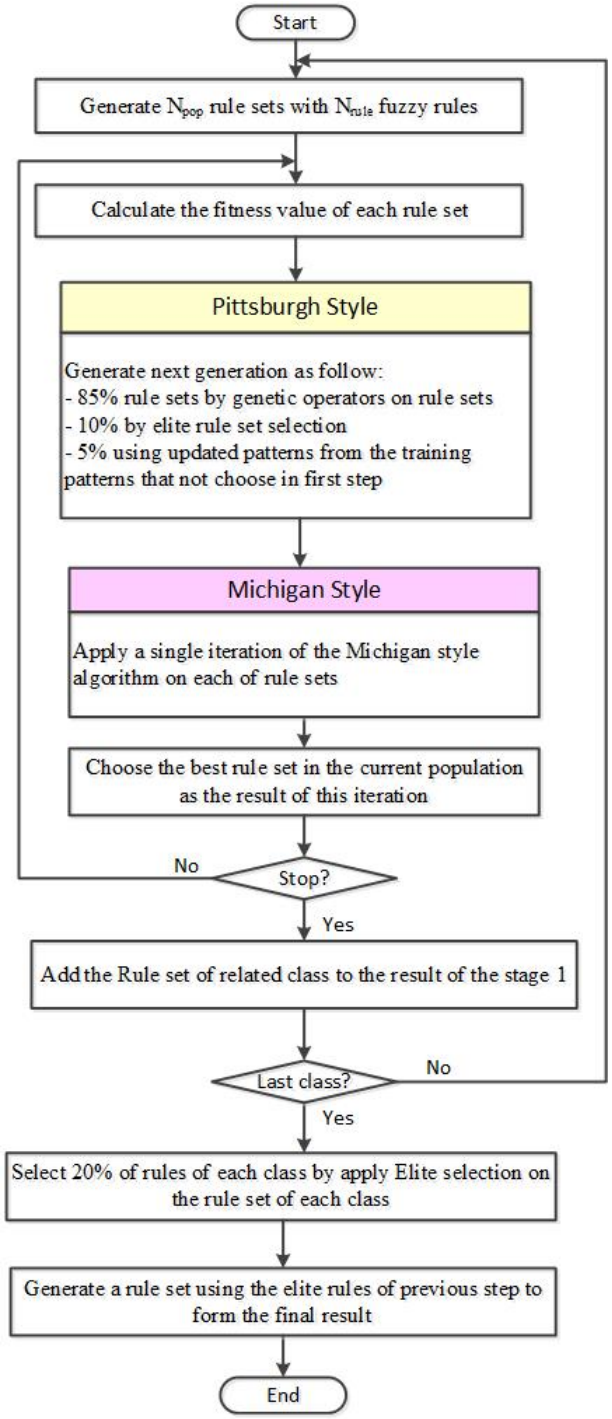

Figure 14: The enhanced IDuFG flowchart

name this approach as Hybrid approach in this paper. In this section, first, we discuss about the dataset that we use during our tests. After that, the results of the algorithm simulation will be discussed.

4.1 Dataset

As we mention in previous sections, one of the important challenging points in designing IDS , is dataset. It is important to use a good dataset for the test phase. The dataset contains features of normal and abnormal behavior use for designing and test intrusion detection systems. There are two ways to model IDS. The first is to create your own dataset and the second is using the existing datasets such as Knowledge Discovery and Data Mining (KDD). Many researchers used KDD CUP 1999 dataset (KDD), but Mahoney and Chans [20] criticized the dataset validity. As they claim information in the dataset does not look like a real traffic in many aspects. Based on their analysis, the IDS created by using KDD with low false alarm may generate high false alarm in real environment. The best advantage of using the existing datasets is that we can evaluate our work with others; but actually because none of the existing datasets contains all kind of todays attacks it is recommended to create your own dataset with as many as possible kind of attacks. Also for different websites and web applications we may encounter different attacks, so it is better to gather the normal and abnormal behavior of their users for each one separately.

The IDS we are proposed in this paper use the existing dataset, because it became possible for us to compare our algorithm with others. But we intend to create a web attack dataset in future works. We will start our work with capturing data from a web application in a normal and an attack situation. In the next step we will get the results of this step in form of log files as an input. Then we will extract some features from these logs. According to these features we will create our own dataset.

Previous works that use their own dataset has limitations in accessibility of suitable web application. Also they encounter several administrative limitations. Furthermore the ability of web intrusion multiple classifications cannot be seen in most of them. To do a more weighty work, we will gather the traffic of a real web application with high traffic and use unlimited session time that help us to increase data density and more accuracy. We will test and simulate more attacks on the selected web application to achieve a better traffic.

We design an IDS to protect a specific system, with higher search ability rather than previous works. However, we used an improved version of KDD called NSL-KDD that is available on [2]. We have utilized this dataset, to train and test our algorithm; because it became possible to compare our algorithm with others.

NSL-KDD is a data set for the evaluation of anomaly detection methods. The data set was generated by gathering the network logs. It consists of two parts, train and test. This dataset contains 41 attributes, a field of label that expresses the class of a pattern, and a difficulty level column. Some important attributes are as follow: duration, protocol type, service, source and destination host, error rate, etc. The content of Label column contains twenty one attack types and normal state. The complete list of attributes is available in [6].

We use NSL-KDD data set to be able to compare our approaches with previous works. It is an improved version of KDD cup 99 data set [21], although it has some prob-

lems mentioned in [26], because there is no other public data set in this domain, researchers use it as a benchmark dataset to evaluate their intrusion detection method.

NSL-KDD is a data set containing numeric and non-numeric values; each row of the data set relating to the network traffic, is a pattern. Each pattern consists of 41 attributes [6], that represent like $(x_{p1}, \ldots, x_{p41})$, $p = 1, 2, \ldots$, number of patterns. Before we start using it, we normalized the numerical values of the extracted patterns into numbers in interval [0,1]. In [10] four different schemes of attributes normalization are introduced to preprocess the data for anomaly intrusion detection. As systematical evaluation results show, normalization improves the detection performance and among the introduced schemes, the statistical normalization scheme is the tiptop one for a large dataset. In [7], authors discussed about how to extract fuzzy rules directly from numerical data for the pattern classification.

According to our studies and the subjects mentioned in [11] explicitly, we encounter five classes including four main attack classes and a normal class instead of twenty two classes in NSL-KDD data set. Table 3 shows the detailed attacks in four main classes. Below you can find a brief description of these classes:

1) Normal patterns produced by normal and usual behavior of a user such as visiting a web page, streaming a video and so on.

2) Denial of Service (DoS) class contains attacks that by using normal connections and behavior of a user in a massive amount try to down the system. This kind of attacks makes the system unable to service the authorized users. The patterns with back, land, neptune, pod, smurf, teardrop labels classify as DoS attack. Figure 15 illustrates a sample of DoS attack to a target server.

3) Remote to User (R2L) is a class of attacks that exploit by a bug. These attacks make the remote user to access the account of a local user. This class consists of ftp_write, guess_passwd, imap, multihop, phf, spy, warezclient, warezmaster labels.

4) User to Root (U2R) class uses some vulnerability that makes the user to be able to gain the root access. Buffer_overflow, perl, loadmodule, rootkit labels are in U2R class.

5) Probing attack is any kind of effort for information gathering by scanning the target system or network to find its vulnerabilities. These vulnerabilities help the attacker to intrude the system. Ipsweep, nmap, portsweep, satan are the labels of this class.

In most IDS, all of 41 existing feature in KDD data set is used to detect the intrusion, while some of them are repetitive and impertinent. These useless features lead to a time consuming detection process means the causes lower performance. In [18] S. Lakhina et al introduced

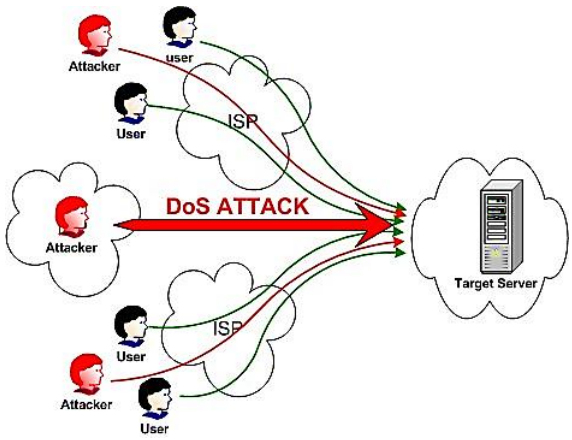

Figure 15: Denial of service attack [3]

Table 3: Different type of attacks in KDD dataset [11]

Four Main Attack Classes	Twenty two Attack Types
Denial of Service (DoS)	Back, land, neptune, pod, smurt, teardrop
Remote to User (R2L)	ftp_write, guess_passwd, imap, multihop, phf, spy, warezclient, warezmaster
User to Root (U2R)	Buffer_overflow, perl, loadmodule, rootkit
Probing	Ipsweep, nmap, portsweep, satan

a new hybrid approach called PCANNA (Principal Component Analysis Neural Network Algorithm) for feature reduction. These approaches reduce the time and memory usage for intrusion detection. It should be mentioned that, in our proposed approaches we use the same feature selection method.

Before using KDD dataset a set of preprocessing actions should be taken on a dataset to prepare it for use. One of the most important actions is normalization. Although there are some anomaly intrusions detection method that does not normalize the data set before test and train. There are several normalization methods but still discussions on their efficiency exist. In [24] four different schemes of attributes normalization are introduced to preprocess the data for anomaly intrusion detection. As systematical evaluation results show, normalization improves the detection performance and among the introduced schemes, the statistical normalization scheme is the tiptop one for a large dataset. We use this method for normalization in our implementations.

4.2 Parameter Setting

There are several parameters in genetic algorithm such as the population size, the mutation rate, the crossover rate, and so on; that influence on the performance and the result of the algorithm. In this subsection, we are going to introduce them briefly and mention the value that we

assign to them in our implementation.

- **Population** Size is one of the important parameters in the genetic algorithm. It says how many individuals are in one generation. If the number of individuals is too few, only a small part of search space is explored and the genetic algorithm has a few possibilities to perform crossover. On the other part, too many individuals, leads the genetic algorithm slow down. We specify 100 rule set that each one contains 20 fuzzy rules as our population.

- **Crossover** Rate indicates the probability of crossover. Simply it states the number of individuals that participate in the crossover. Generally, it should be high, about 0.8-0.95.

- **Mutation** Rate is the probability of alteration of attributes of an individual in the population. It is usually a small value between 0.01-0.1. Of course it could be in 0-1 interval but the mentioned rate is the best average in general.

- W_t is the weight of the correct classification. This value is used in fitness calculation. Actually the number of patterns that classify accurately is multiple by this weight.

- W_f is the weight of the incorrect classification. This value is also used in fitness calculation. Actually the number of patterns that classify erroneously is multiple by this weight and consider as a penalty for the related fuzzy rule.

According to the similar related work, we set the mentioned parameters in our implementation. Table 4 summarizes these parameter values. These parameters are set according to the previous works and experimental results.

Table 4: Parameters value in computer simulation

Parameter	Value
Number of rule sets	100
Number of rules in each rule set	20
Number of Generations	50
Crossover probability	0.9
Mutation probability	0.1
w_t	0.1
w_f	0.9

4.3 Simulation Results

In this section, we test our proposed approaches on NSL KDD; in the following subsections, first we introduce the assessment measures, then we explain the results of the test for each approach.

4.3.1 Assessment Measures

To assess the proposed approaches we use different measures that have been used in most literatures for intrusion detection evaluation [17]; these measures are Recall, Precision, F-measure, and accuracy. The important factors in IDS are maximizing the accuracy while minimizing the False alarms.

Recall is the fraction of Correctly Classified Instances or p(p), to the total number of input patterns that must have been classified correctly. In other words, recall is a fraction of True Positive Rates to the number of all cases that should have been classified as positive. Recall can be defined according to the following equation (TP and FN indicate True Positive and False Negative):

$$Recall = \frac{TP}{TP + FN}.100. \tag{8}$$

Precision is the fraction of true positive instances to all positive instances including those positive instances as classified by the algorithm. In other words, precision is the number of correct results divided by all the results that have been specified by algorithm. Precision then can be seen as a measure of exactness while recall can be seen as a measure of completeness (FP indicates False Positive).

$$Precision = \frac{TP}{TP + FP}.100. \tag{9}$$

F-Measure or F1 is another measure which uses both precision and recall as shown in Figure 16. The value of F measure can show how accurate the algorithm has been, means the higher F shows that an algorithm has been more accurate. The following is the formula for obtaining F score:

$$F_Measure = \frac{2.Recall.Precision}{Precision + Recall}. \tag{10}$$

Accuracy indicates the percentage of closeness of the obtained value to the actual value of the algorithm. It is calculated by the following equation. Figure 17 illustrates the difference between accuracy and precision.

$$Accuracy = \frac{TP + TN}{p(p) + p(n).} \tag{11}$$

4.4 Basic Approach

We compare our basic approach with the Hybrid approach, since it is the nearest work to ours. We find the measures introduced above for each of the approaches and briefly describe them in follow. These measures have been used in literatures to compare and assess many proposed algorithms.

The True Positive Rate of Basic IDuFG is 98.052 and the False Positive Rate is 0.511; while in case of Hybrid approach introduced by Ishibuchi in [14], True Positive Rate is 97.865 and the False Positive Rate is 0.7251. This shows that Basic IDuFG has been better able to report a case as positive where it has actually been positive

Figure 16: F-measure definition [17]

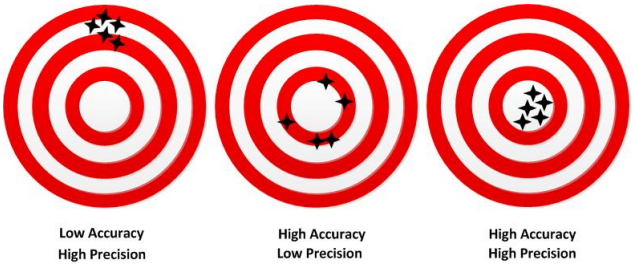

Figure 17: Accuracy and precision

compared to Hybrid approach. It also shows that Basic IDuFG has been less likely to report a case as positive when in fact it has not been positive compared to the Hybrid approach.

The recall rate for Basic IDuFG in the dataset for this experiment is 99.481 and for Hybrid approach is 99.256. According to the definition of recall this means that among all the instances that should have been classified as positive, Basic IDuFG has been able to spot a large number of them as positive while Hybrid approach has not been very successful in differentiating the positive instances.

Since F-measure shows the accuracy of an algorithm, the higher F shows that an algorithm has been more accurate. In this research the obtained F value, for Basic IDuFG and Hybrid approach are 99.089 and 98.420 respectively. This suggests that Basic IDuFG has had a higher accuracy when compared to Hybrid approach. Table 5 shows a comparison between the hybrid GBML that introduce in previous work by Ishibushi and two proposed approaches.

To examine the classification accuracy, first we run the algorithm for each class separately and generate twenty fuzzy rules for each class. Then we check the classification accuracy of this rules for each class separately. The processes of the Classification Accuracy of each class are presented separately in Figures 18 to 20.

The comparison results between hybrid, basic and enhanced approaches are shown in 5. In comparison with hybrid approach that introduced by Ishibuchi, Our basic approach has higher classification accuracy and less false alarm ration while supporting multiple attack classifications, and faster convergence. The enhance approach has a tradeoff between computational time and classification accuracy. In this approach with a little decline in classification accuracy, we reduce the computational time of test phase down to 80%. We can find out that the algorithm behaves differently for the pattern of different classes, but in all of them it converges to the maximum result at most in 50 iterations.

It represent that our proposed algorithm converges to the maximum detection rate on NSL KDD data set faster than the previous hybrid algorithm introduced in [15]. Faster convergence leads to reduction in the run time.

Table 5: Comparison results between hybrid, basic, and enhanced approaches

Measure	Hybrid GBML	Basic IDuFG	Enhanced IDuFG
True Positive	97.865	98.052	97.4
False Negative	0.725	0.511	0.63
Recall	99.256	99.481	99.357
Precision	97.598	98.701	98.54
F-Measure	98.42	99.089	98.701
Classification Accuracy	97.891	98.199	97.437

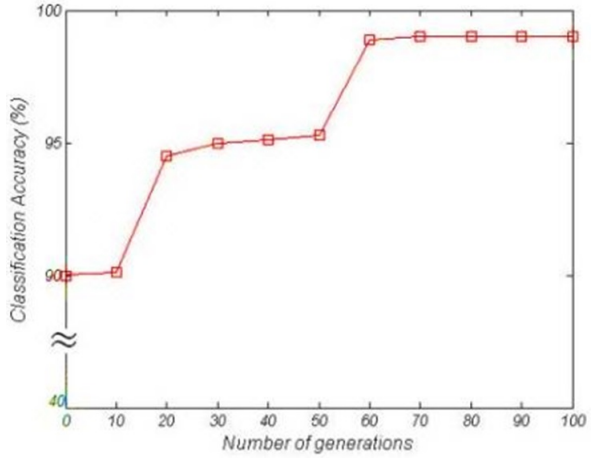

Figure 18: Classification accuracy of basic IDuFG for normal patterns

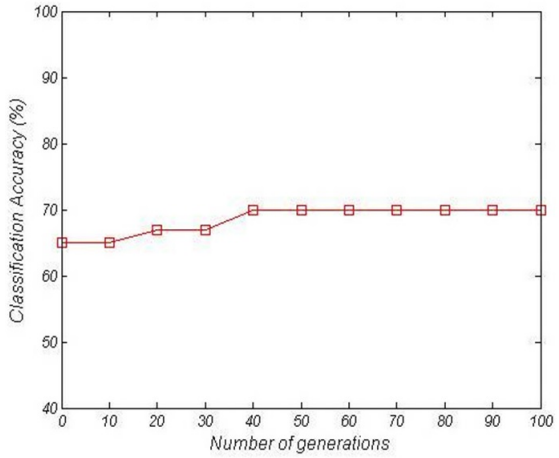

Figure 19: Classification accuracy of basic IDuFG for U2R patterns

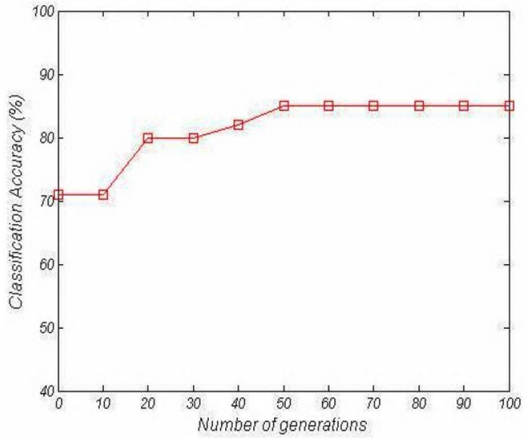

Figure 20: Classification accuracy of basic IDuFG for R2L patterns

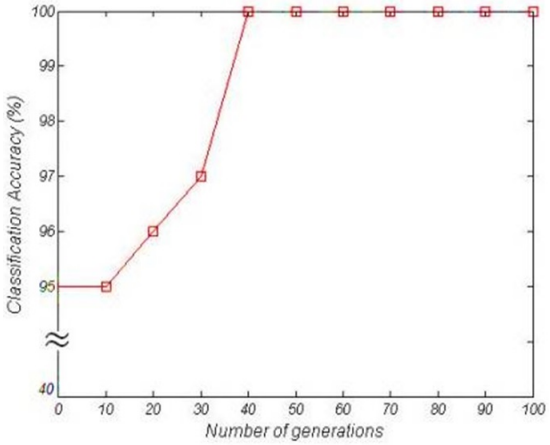

Figure 21: Classification accuracy of basic IDuFG for DOS patterns

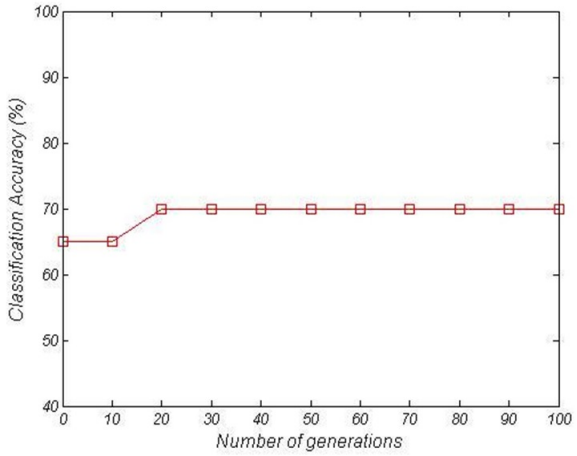

Figure 22: Classification accuracy of basic IDuFG for PRB patterns

We can find out that the algorithm behaves differently for the pattern of different classes, but in all of them it converges to the maximum result at most in 50 iterations.

As it is obvious in these figures, the algorithm behaves differently for the patterns of each class. The reason of this behavior is related to different number of patterns of each class and the relevant feature for each class. For example in DOS class that has the high number of training patterns, we get better classification accuracy.

Table 6: Experimental results of basic approach for each class

Class	Classification Accuracy (%)	False Alarm (%)
Normal	98.5	0.0019
DOS	99.2	0.0047
R2L	89	0.056
U2R	72.3	0.7
PRB	73.8	0.2

Since the number of training patterns of PRB attacks are approximately less than others, we get less classification accuracy as you see in Figure 22. As number of final fuzzy rules is equal to one hundred, in test phase we should compare a pattern with each of them and determine its corresponding class. So the computational time in the test phase is long in comparison with the enhanced approach. The summary of classification accuracy and false alarm rates for each class is shown in Table 6.

First we run the algorithm for each class separately and generate twenty fuzzy rules for each class. Then we check the detection rate of this rules for each class separately. To see the total classification accuracy of these fuzzy rules in basic approach, we check the classification accuracy of these rules for the test patterns of all classes. An average experimental result of over 20 runs is shown in Figure 23 for all classes together. It represents that our proposed algorithm converges to the maximum classification accuracy on NSL KDD data set faster than the hybrid algorithm introduced by [15]. Faster convergence leads to reduction in the run time.

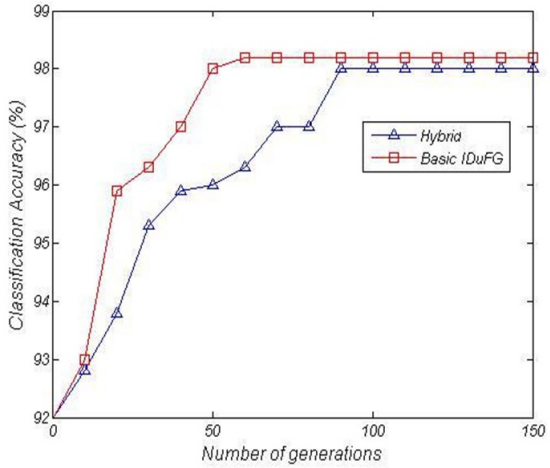

Figure 23: Total classification accuracy of basic IDuFG and hybrid approach

4.5 Enhanced Approach

In order to reduce the computational time in the test phase, we reduce the number of fuzzy rules in the obtained population. As we mentioned above we achieve one hundred fuzzy rules. In the second approach, we decrease the number of fuzzy rules by elite selection of the rule of each class. We select the rules as their certainty grade, the fuzzy rules with the highest certainty is selected in each class. Therefore we have twenty fuzzy rules with the highest certainty grades. The result of testing these rules is shown in Figure 24.

As the experimental results show, although the computational time in the test phase reduces because of the reduction of the fuzzy rules number, the classification accuracy is also reduces a little bit. This is because of the fact that we omit several rules with less certainty grade, but they were able to detect some patterns but not the existing rules. Enhanced IDuFG approach with a little decline of the classification accuracy the computational time in the test phase of the algorithm reduces about 20% of the needed time for the basic approach. Comparison of computational time of these two approaches is shown in Figure 24.

The computational time in the test phase is directly depends on the number of rules. In this approach, it deceases up to 80%, because the number of fuzzy rules is decreased up to 80%. The result is shown in Figure 25. As the test phase time is a small part of the total time (including test and train phase), it influences on the total computational time less than the test phase computational time.

In enhanced IDuFG approach, with a little decline of the classification accuracy, the computational time in test phase of algorithm reduces about 5% of the time needed for basic approach. The comparison of the total computational time of these two approaches is shown in Figure 26.

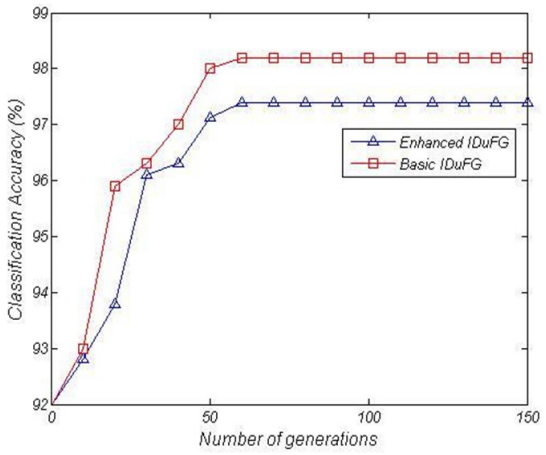

Figure 24: Classification accuracy of basic and enhanced IDuFG

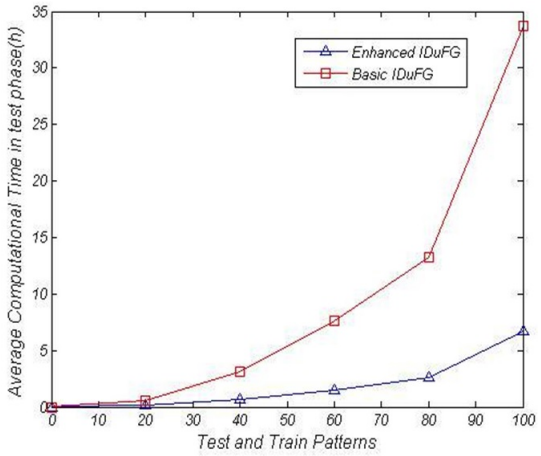

Figure 25: Test phase computational time reduction

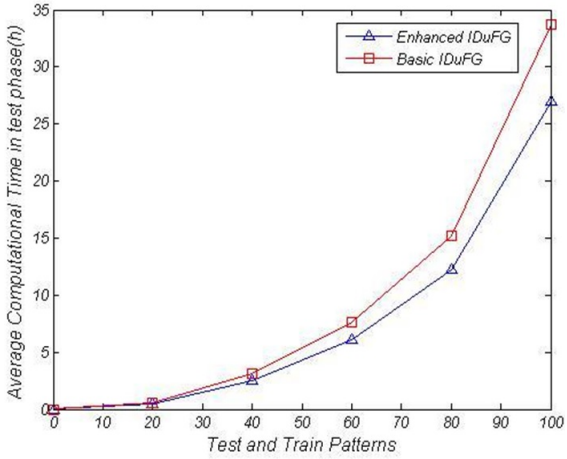

Figure 26: Total computational time reduction

5 Conclusions

Our Proposed system called IDuFG, is an IDS which uses hybrid Fuzzy Genetic algorithm. It supports multiple attack classifications. Two proposed approaches of its algorithm are basic and enhanced. The efficiency of algorithm is kept by using Elitism concept while try to improve it by using more random patterns. The number of generated rules in the enhanced approach is 20% of the basic approach rules. With a little decline of classification accuracy the computational time in the test phase is shorter than the basic approach. So the number of generated rules in the enhanced approach is 80% less than the basic approach, and as a result the computational time of the test phase reduces 80% while the total computational time will be reduced about 5%. With a little decline of the classification accuracy the computational time in the test phase is shorter than the basic approaches.

Finally, as future work we plan to create a dataset consisting of web normal and attack traffic and testing the proposed fuzzy GBML algorithm on the web traffic dataset. Also Utilize Immune systems approach to improve the performance of web anomaly intrusion detection

systems.

Acknowledgments

Hereby we express our special thanks to Mr. Iman Khalkhali as Web-based Anomaly Detection Lab (WADL) supervisor for his supports during the tests.

References

[1] http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.
[2] http://iscx.ca/NSL-KDD.
[3] http://www.crazyengineers.com/threads/dos-attack-possibility-on-iran-by-anonymous.63622/.
[4] M. S. Abadeh, J. Habibi, Z. Barzegar, and M. Sergi, "A parallel genetic local search algorithm for intrusion detection in computer networks," *International Journal of Engineering Applications of Artificial Intelligence*, vol. 20, no. 8, pp. 1058–1069, 2007.
[5] M. S. Abadeh, J. Habibi, and E. Soroush, "Induction of fuzzy classification systems using evolutionary aco-based algorithms," in *Proceedings of First Asia International Conference on Modeling and Simulation*, pp. 346–351, Phuket, 2007.
[6] M. S. Abadeh, H. Mohamadi, and J. Habibi, "Design and analysis of genetic fuzzy systems for intrusion detection in computer networks," *International Journal of Expert Systems with Application*, vol. 38, pp. 7067–7075, 2011.
[7] S. Abe and M.-S. Lan, "A method for fuzzy rules extraction directly from numerical data and its application to pattern classification," in *IEEE Transactions on Fuzzy Systems*, vol. 3, pp. 18–28, Ibaraki, 1995.
[8] M.E. Cintra and H. de Arruda Camargo, "Fuzzy rules generation using genetic algorithms with self-adaptive selection," in *IEEE International Conference on Information Reuse and Integration*, pp. 261–266, Las Vegas, 2007.
[9] O. Cordon, F. Herrera, F. Hoffmann, and L. Magdalena, *Genetic Fuzzy Systems*, vol. 19. World Scientific Publishing Publication, 2001.
[10] R. Ensafi, S. Dehghanzadeh, and M. R. Akbarzadeh, "Optimizing fuzzy k-means for network anomaly detection using pso," in *Proceeding of IEEE International Conference on Computer Systems and Applications*, pp. 686–693, Doha, 2008.
[11] D. M. Farid, N. Harbi, E. Bahri, M. Z. Rahman, and C. M. Rahman, "Attacks classification in adaptive intrusion detection using decision tree," in *World Academy of Science, Engineering and Technology*, pp. 86–90, 2010.
[12] A. Fernandez, S. Garcia, J. Luengo, E. Bernado Mansilla, and F. Herrera, "Genetics-based machine learning for rule induction: State of the art, taxonomy, and comparative study," in *IEEE Transaction on Evolutionary Computation*, pp. 913–941, Spain, 2010.
[13] L. Hui and C. Yonghui, "Research intrusion detection techniques from the perspective of machine learning," in *Proceedings of the Second International Conference on Multimedia and Information Technology*, pp. 166–168, Kaifeng, 2010.
[14] H. Ishibuchi and T. Nakashima, "Improving the performance of fuzzy classifier systems for pattern classification problems with continuous attributes," in *IEEE Transactions on Industrial Electronics*, vol. 46, pp. 1057–1068, 1999.
[15] H. Ishibuchi, T. Yamamoto, and T. Nakashima, "Hybridization of fuzzy gbml approaches for pattern classification problems," in *IEEE Transaction on Systems, Man, and Cybernetics*, vol. 35, pp. 359–365, Japan, 2005.
[16] V. Jazayeri, J. Habibi, M. S. Abadeh, and P. Pirzadeh, "Network attacks detection rules extraction using combination of pittsburgh and michigan," in *11th International CSI Computer Conference*, pp. 24–26, Iran, 2006.
[17] I. Khalkhali, R. Azmi, M. Azimpour, , and M. Khansari, "Host-based web anomaly intrusion detection system, an artificial immune system approach," *International Journal of Computer Science Issues*, vol. 8, no. 5, pp. 14–24, 2011.
[18] S. Lakhina, S. Joseph, and B. Verma, "Feature reduction using principal component analysis for effective anomaly-based intrusion detection on nsl-kdd," *International Journal of Engineering Science and Technology*, vol. 2, no. 6, pp. 1790–1799, 2010.
[19] Y. Li, L. Guo, Z. H. Tian, and T.B. Lu, "A lightweight web server anomaly detection method based on transductive scheme and genetic algorithms," *International Journal of Computer Communication*, vol. 31, pp. 4018–4025, 2008.
[20] M.V. Mahoney and P. K. Chani, "An analysis of the 1999 darpa/lincoln laboratory evaluation data for network anomaly detection," in *Proceedings of Recent Advances in Intrusion Detection Conference*, pp. 220–237, Ottawa, 2003.
[21] J. McHughi, "Testing intrusion detection systems: a critique of the 1998 and 1999 darpa intrusion detection system evaluations as performed by lincoln laboratory," in *ACM Transactions on Information and System Security*, vol. 3, p. 262–94, 2000.
[22] Y. Nojima, H. Ishibuchi, and I. Kuwajimai, "Comparison of search ability between genetic fuzzy rule selection and fuzzy genetics-based machine learning," in *Proceedings of International Symposium on Evolving Fuzzy Systems*, pp. 125–130, Ambleside, 2006.
[23] C.H. Tsang, S. Kwong, and H. Wang, "Genetic-fuzzy rule mining approach and evaluation of feature selection techniques for anomaly intrusion detection," *International Journal of Pattern Recognition Society*, pp. 2373–2391, 2007.
[24] W. Wang, X. Zhang, S. Gombault, and S. J. Knapskog, "Attribute normalization in network intrusion

detection," in *10th International Symposium on Pervasive Systems, Algorithms, and Networks*, pp. 448–453, Kaohsiung, 2009.

[25] S.X. Wu and W. BanZhaf, "The use of computational intelligence in intrusion detection systems," *International Journal of Applied Soft Computing*, vol. 10, pp. 1–35, 2009.

[26] H. Zhang, B. Zhang, and F. Wang, "Automatic fuzzy rules generation using fuzzy genetic algorithm," in *Sixth International Conference on Fuzzy Systems and Knowledge Discovery*, pp. 107–112, Tianjin, 2009.

Ghazaleh Javadzadeh received her BS degree in Computer Software engendering from Islamic Azad University- South Tehran Branch, Iran in 2009 and her MS degree in Information Technology from Sharif University of Technology-International Campus, Kish, Iran in 2012. She was also a member of Society Engineering of Computer in Islamic Azad University from 2007 to 2009, and Web-based Anomaly Detection Lab (WADL) in 2012. She was a researcher in Security Research Faculty of ICT Research Institute (ITRC), Tehran, Iran in 2012. She is an instructor at University of Applied Science and Technology, Tehran, Iran. Her main research interests are network and web security, and software security.

Reza Azmi received his BS degree in Electrical Engineering from Amirkabir university of technology, Tehran, Iran in 1990 and his MS and PhD degrees in Electrical Engineering from Tarbiat Modares university, Tehran, Iran in 1993 and 1999 respectively. Since 2001, he has joined Alzahra university, Tehran, Iran. He was an expert member of Image Processing and Multi-Media working groups in ITRC (From 2003 to 2004), Optical Character Recognition working group in supreme council of information and communication technology (From 2006 to 2007) and Security Information Technology and Systems working groups in ITRC (From 2006 to 2008). He was Project Manager and technical member of many industrial projects. Dr Azmi is founder of Operating System Security Lab (OSSL), Medical Image Processing Lab (MIPL), Face and Facial Expression Recognition Lab (FFERL), Web-based Anomaly Detection Lab (WADL) and Optical Character Recognition Lab (OCRL) in Alzahra University. He is currently an Assistant Professor of Computer Engineering at Alzahra University.

A Dynamic Threshold Decryption Scheme Using Bilinear Pairings

Brian King
Department of Electrical and Computer Engineering, Indiana University - Purdue University Indianapolis
723 West Michigan Street, SL160, Indianapolis, IN 46202, USA
(Email: briaking@gmail.com)

(Received May. 15, 2013; revised and accepted Jan 13 & Feb. 6, 2014)

Abstract

A dynamic threshold sharing scheme is one that allows the set of participants to expand and contract. In this work we discuss dynamic threshold decryption schemes using bilinear pairing. We discuss and analyze existing schemes, demonstrate an attack and construct a significantly more efficient secure scheme.

Keywords: Bilinear pairing, dynamic threshold scheme, ECC

1 Introduction

Secret sharing is a mechanism that is used to share out a secret to multiple parties such that only those authorized sets are allowed to recover the secret key. Threshold secret sharing is an example of a secret sharing scheme, where the authorized sets consists of those groups of participants whose membership is greater than or equal to the threshold. A t out of n threshold sharing [2, 10] is such that any set of participants that contains t or more are authorized and can recover the secret. The most common use of threshold secret sharing is to build threshold cryptographic applications. Threshold cryptography refers to a technique where threshold secret sharing is used to compute a function of the secret rather than the secret itself. Examples of functions/applications would include a decryption of ciphertext and signature schemes. Threshold cryptography has been used to describe many group oriented applications [6].

Today, it is common security technique that is used to achieve computationally secure group access. A dynamic threshold sharing scheme [8] is threshold sharing scheme where the participant set is dynamic, allowing it to expand, as well as contract. Identity based encryption is a technique such that some public identity information is used as a public key. Identity based encryption was first proposed by Shamir [11]. In [3], Boneh and Franklin constructed an identity based encryption scheme based on bilinear pairings.

In this paper we discuss a dynamic threshold encryption scheme, we discuss two current schemes and discuss attacks in their schemes. We then provide an improved dynamic threshold decryption scheme. We assume we have the following system. Users enroll in a encryption/decryption scheme. Once enrolled their identification ID is registered. The service of the system is such that users can register their identity to a trusted third party, denoted by TTP, who then constructs and publishes their identity-based-public-key. The precise process of the registration is outside the scope of our work.

The user can then have messages encrypted to them based on the system public key and their identity and then have threshold servers decrypt the ciphertext for the user. The process in which the user makes the request for a decryption is outside the scope of the paper. The servers are dynamic in nature and can grow and contract over time.

1.1 Dynamic Threshold Decryption Scheme

The concept of a *dynamic threshold decryption scheme* [8], is such that a public key encryption scheme exists, the decryption key `sk` is shared out to a set of n decryption servers, denoted by $\Gamma_1, \Gamma_2, \ldots, \Gamma_n$, in such a way that any t out of n can decrypt a message which is encrypted using the public key `pk`, and the membership of these servers is dynamic in nature.

The goals for a dynamic dynamic sharing scheme [8] are:

- The system can refresh the decryption key without having to modify any of the shares of the decryption servers.
- If the system adds a new decryption server then the systems does not have to modify the decryption key

nor do they have to modify any of the other decryption server's shares.

- The removal of a decryption server does not require the modification of any other decryption server's shares, further it does not require the modification of the decryption key.

- A decryption server can refresh its private key without requiring any other decryption server to modify their private key, further this refresh does not require the system to modify of any other decryption server shares.

In order to develop an enhanced system we propose a revised set of criteria, to support these revised features, additional public information will be available in a publicly available setting, such as a bulletin board. This additional bulletin board information is typically used to support a set of authorized servers ability to reconstruct the decryption key and/or to decrypt a message for a user.

In [5], Chen, Gollman, Mitchell and Wild introduced the concept of reusable polynomials for secret sharing with a goal of supporting dynamic thresholds. The main properties that they were interested in were:

Perfect security or computational security. A secret sharing scheme is perfectly secure if unauthorized subsets of shareholders cannot obtain information about the secret. A scheme is computationally secure provided it is computationally infeasible to determine the secret from an unauthorized subset.

Verifiability. First, each shareholder should be able to verify their received share to detect a dishonest or faulty dealer. Secondly, during secret reconstruction a forged share contributed by a cheating shareholder can be detected by the other shareholders.

Online shareholders. Shareholders can dynamically join or leave the sharing group without having to redistribute new shares secretly to the existing shareholders.

Reusable shares. Shares need to be reusable even after the shared secret has been reconstructed.

Our criteria for secure dynamic threshold sharing:

- The system can refresh decryption key without having to contact and/or send new shares to any of the decryption servers.

- When the system adds a new decryption server then they do not have to modify the decryption key nor do they have contact and/or send new shares to any of the other decryption server's shares.

- The removal of a decryption server does not require contact and/or the sending new shares of any other decryption server's shares, further it does not require the modification of the decryption key.

- A decryption server can refresh its private key without requiring the any other decryption server to modify their private key, further it does not require contact and/or sending new shares to any other decryption server shares.

- Each server should be able to verify their shares compute the secret.

- All system/shareholder (server) computations are efficient.

This set of criteria allows for the use of public setting modification (for example, shares placed on a public bulletin board).

1.2 Bilinear Pairings

A mathematical tool that we utilize in our work will be the bilinear pairing. Let $\mathbb{G}$, and $\mathbb{G}_1$ be cyclic groups of prime order p, such that both $\mathbb{G}$ and $\mathbb{G}_1$ are multiplicative groups.

Definition 1. *A map $e : \mathbb{G}\times\mathbb{G} \to \mathbb{G}_1$ is said to be bilinear pairing if it has the following properties:*

Bilinearity. $e(g^a, w^b) = e(g, w)^{ab}$ *for all* $g, w \in \mathbb{G}$ *and* $a, b \in \mathbb{Z}_p^*$.

Non-degeneracy. $e(g, g) \neq 1$ *in* $\mathbb{G}_1$ *and* $g \neq 1$ *in* $\mathbb{G}$.

Computability. *There exists an efficient algorithm that computes* $e(g, w)$ *for all* $g, w \in \mathbb{G}$.

We will assume that the *discrete log problem* (DLOG) is hard. The DLOG problem is such that given group $\mathbb{G}$, and $g \in \mathbb{G}$ and g^k it is s "computationally infeasible" to determine k. We will assume that the discrete log problem is "hard" in $\mathbb{G}_1$.

In this paper we will be using a bilinear map e. We assume that the discrete log problem is "hard" even in the presence of the bilinear map, that is, given generator g, the value g^a and the pairing map e, it is "hard" to compute the exponent a. We will assume that *Computational Diffie-Hellman (CDH)* problem is also 'hard" in the presence of the bilinear map, thus given g^a, g^b, and $e(g^a, g^b)$ it is hard to compute ab. We will assume that the CDH problem is hard in $\mathbb{G}_1$. The *Decision Diffie-Hellman (DDH)* problem is the problem concerning whether one can distinguish between (g, g^a, g^b, g^c) and (g, g^a, g^b, g^{ab}). The *Decision Diffie-Hellman (DDH)* problem is "easy" in $\mathbb{G}_1$ due to the existence of the bilinear map e. Consequently, we will be working in an algebraic setting described by Boneh et al. [4] as the *Gap Diffie-Hellman*

(GDH) group. The group $\mathbb{G}_1$ is called a GDH group if DDH is easy in $\mathbb{G}_1$ but CDH is hard. Thus $\mathbb{G}_1$ is a GDH group.

1.3 Identity-based Encryption

The concept of identity-based encryption was proposed by Shamir in 1984 [11]. The construction of an identity-based encryption scheme was an open problem until solved by Boneh and Franklin in [3]. Today there are a number of identity based encryption schemes proposed, we refer the reader to a survey of the schemes [1].

2 Construction of Dynamic Threshold Decryption Scheme from Pairing

In this section we discuss a dynamic threshold decryption scheme proposed by Long and Chen [9]. Unfortunately there is a typographical error in their work, thus placing a level of ambiguity to their scheme. In [7] Kim, Lim, Yie, and Kim analyzed the Long scheme, because of the typographical error they had to make an interpretation of the error, their interpreted scheme was flawed. In Kim et al.'s cryptanalysis, they showed that their interpretation of the Long et al. scheme is insecure. Long and Chen constructed their *dynamic threshold decryption scheme* using bilinear pairings. Their scheme attempts to solve the problem of decrypting the ciphertext without compromising the master key, and was inspired by [12]. The Long et al. scheme is summarized in the following steps:

Setup. There is a trusted party private key generator (TTP) which chooses two bilinear groups $\mathbb{G}_1$ and $\mathbb{G}_2$, where each group has prime order p. Let g be a generator of $\mathbb{G}_1$, and $e : \mathbb{G}_1 \times \mathbb{G}_1 \to \mathbb{G}_2$ a bilinear map. The messages are denoted by M. Assume all messages M belong to $\mathbb{G}_1$. We assume that each user u has a public key ID_u that belongs to $\mathbb{Z}_p^*$. The TTP selects random $x, y \in \mathbb{Z}_p^*$ and computes $X = g^x, Y = g^y$. The public parameters are denoted by $cp = (g, X, Y)$ and the master key $mkey = (x, y)$ where x remains secret and y is secret but renewed periodically.

KeyGen. Initially there are n decryption servers $\Gamma_1, \Gamma_2, ..., \Gamma_n$. Each server Γ_i possesses a secret key $s_i \in \mathbb{Z}_p^*$ and a corresponding public key $P_i = g^{s_i}$. The TTP selects a random polynomial $f(z)$ of degree $t-1$ over $\mathbb{Z}_p^*$ by selecting $b_1, b_2, \ldots b_{t-1}$ from $\mathbb{Z}_p^*$, the polynomial $f(z)$ satisfies $f(z) = y + \sum_{i=1}^{k-1} b_i z^i$, here $b_{k-1} \in \mathbb{Z}_p^*$. For each i, the TTP computes

$$k_i = g^{\frac{f(i)}{(ID+x)P_i^y}}$$

and $v_i = e(g, g)^{f(i)}$. The TTP publishes k_i and v_i on the public available site (which we will call the bulletin board).

Encryption. Suppose *Alice* would like to transmit message M to *Bob* privately. She gets *Bob*'s identification, denoted by ID, as well as the *master TTP keys* X and Y. She then encrypts message M with public key ID, by picking up a random $S \in \mathbb{Z}_p^*$ and computes the ciphertext C by $C = (g^{S \cdot ID} \cdot X^S, e(g, Y)^S \cdot M) = (A, B)$.

Γ_i 's Sub-decryption. After Bob receives the decrypted message that was sent by Alice, Bob can ask the servers to decrypt it. This can be achieved whenever t decryption servers $\Gamma_{i_1}, \ldots, \Gamma_{i_t}$ cooperate and reconstruct the message by utilizing their shares (on the bulletin board) of the decryption key in the t of n threshold sharing scheme. In this step, we illustrate how a server Γ_i calculates its decryption share δ_i of the ciphertext, which is computed with the use of the server's private key s_i. According to [9], the decryption server Γ_i calculates the share δ_i by computing

$$\delta_i = e(A, g^{Y^{s_i}})^{k_i} = \cdots = e(g, g)^{S \cdot f(i)}. \qquad (1)$$

Note: *The formula given in Equation (1) is wrong!* We address this issue in Section 2.1.

Decryption. Assuming decryption servers $\Gamma_1, \Gamma_2, ..., \Gamma_t$ want to decrypt the ciphertext, each server Γ_i computes δ_i and sends it to the combiner who computes Δ by:

$$\begin{aligned}
\Delta &= \prod_{j=1}^{t} (\delta_j)^{\prod_{\substack{i=1 \\ i \neq j}}^{t} \frac{-i}{j-i}} \\
&= \prod_{j=1}^{t} e(g, g)^{sf(j) \prod_{\substack{i=1 \\ i \neq j}}^{t} \frac{-i}{j-i}} \\
&= e(g, g)^{\sum_{j=1}^{t} sf(j) \prod_{\substack{i=1 \\ i \neq j}}^{t} \frac{-i}{j-i}} \\
&= e(g, g)^{sy} \\
&= e(g, Y)^s.
\end{aligned}$$

Note: *The above calculation only make sense provided that* $\delta_j = e(g, g)^{S \cdot f(j)}$. Again this is addressed in Section 2.1.

If Δ equals $e(g, Y)^S$, then M can be recovered by computing $M = B \cdot \Delta^{-1}$. The use of the bulletin board allows Long et al. to achieve the revised dynamic properties. The share v_i can be used by each server Γ_i to verify the

correctness of the share k_i, a property we are interested in satisfying. Further any server Γ_j can verify the correctness of the shares $v_1, \ldots v_n$ by selecting any t of them and computing

$$e(g,g)^y \stackrel{?}{=} \prod_{i=1}^{t} v_{w_i}^{\prod_{i=1, i\neq j}^{t} \frac{-w_j}{w_i - w_j}}.$$

Long et al. claimed that their scheme satisfied the following dynamic threshold requirements.

TTP refreshes secret key. In the case a new secret key y_{new} is selected, then $Y_{new} = g^{y_{new}}$ is computed and a new polynomial f_{new} is selected, new shares $k_{i,new}$ and $v_{i,new}$ will be shared out to the servers $\Gamma_1, \ldots, \Gamma_n$.

TTP adds new decryption server. In the case the TTP adds a new server Γ_{n+1}, they simply use the polynomial value $f(n+1)$ and generate a new share

$$k_{n+1} = g^{\frac{f(n+1)}{(ID+x)P_{n+1}^y}} \text{ and } v_{n+1} = e(g,g)^{f(n+1)}$$

here P_{n+1} is Γ_{n+1}'s public key.

TTP removes a decryption server. Assume without loss of generality that server Γ_n is dismissed then a new polynomial f_{new} is selected with the same secret key y and new shares $k_{i,new}$ and $v_{i,new}$ will be shared out to the bulletin board by the TTP for the servers $\Gamma_1, \ldots, \Gamma_{n-1}$.

Server Γ_i refreshes their secret key s_i. If Γ_i refreshes their secret key and select $s_{i,new}$ then they will compute $P_{i,new} = g^{s_{i,new}}$ and new shares $k_{i,new}$ and $v_{i,new}$ will be shared out to the bulletin board by the TTP for the servers.

Remarks: Clearly this scheme does not possess the security (nor correctness) that the authors claim. More importantly, this is very inefficient. In reality the labelling of shares as k_i and v_i is inaccurate as they depend not only on the server Γ_i but also on the user's identification ID. That is, if there are m users $\{ID_1, ID_2, \ldots, ID_m\}$ then there are m distinct (k_i, v_i) pairs (as illustrated below–one pair for each user ID). That is, we have

	Γ_1	Γ_2	$\cdots$	Γ_n
ID_1	(k_{1,ID_1}, v_{1,ID_1})	(k_{2,ID_1}, v_{2,ID_1})	$\cdots$	(k_{n,ID_1}, v_{n,ID_1})
ID_2	(k_{1,ID_2}, v_{1,ID_2})	(k_{2,ID_2}, v_{2,ID_2})	$\cdots$	(k_{n,ID_2}, v_{n,ID_2})
$\vdots$	$\vdots$		$\ddots$	
ID_m	(k_{1,ID_m}, v_{1,ID_m})	(k_{2,ID_m}, v_{2,ID_m})	$\cdots$	(k_{n,ID_m}, v_{n,ID_m})

Thus the cost of executing the refresh properties are (in big O notation) is described in Table 1.

Table 1: Computational cost of Long et al. scheme

operation	computational cost
TTP refreshes secret key	$O(mn)$
TTP adds new decryption server	$O(m)$
TTP removes a decryption server	$O(mn)$
server Γ_i refreshes their secret key	$O(m)$

2.1 Kim et al.'s Interpretation of the Long Scheme

Clearly there is a typographical error in Long et al. scheme. In [7] Kim, Lim, Yie, and Kim cryptanalyzed the Long et al. scheme. Unfortunately due to the typographical error in the Long et al. paper [9], Kim et al. [7] had to interpret the scheme, they interpreted the Long scheme as follows:

Setup. *Same as before.*

KeyGen. *Same as before*, exception: for each i, the TTP computes $k_i = g^{\frac{f(i)}{(ID+x)p_i^y}}$, $v_i = e(g,g)^{f(i)}$ and publishes k_i, v_i.

Encryption. *Same as before.*

Γ_i 's Sub-decryption. Bob can receive the message sent by Alice by having t servers $\Gamma_{i_1}, \ldots, \Gamma_{i_t}$ reconstruct the message by utilizing their shares of the decryption key in the t of n threshold sharing scheme. In this step, the server Γ_i calculates its decryption share δ_i of the ciphertext as follows:

$$\begin{aligned} \delta_i &= e(A, k_i \cdot Y^{s_i}) \\ &= e(g,g)^{S\cdot(ID+x)\cdot\frac{f(i)}{ID+x}} \\ &= e(g,g)^{S\cdot f(i)}. \end{aligned}$$

The above derivations are correct.

Decryption. Assuming t decryption servers $\Gamma_1, \Gamma_2, ..., \Gamma_t$ wish to decrypt the ciphertext for user ID, one of the servers collects $\delta_1, \delta_2, ..., \delta_t$ and computes Δ as follows:

$$\begin{aligned} \Delta &= \prod_{j=1}^{t} \delta_j^{\prod_{\substack{i=1 \\ i\neq j}}^{t} \frac{-i}{j-i}} \\ &= \prod_{j=1}^{t} e(g,g)^{sf(j)\prod_{\substack{i=1 \\ i\neq j}}^{t} \frac{-i}{j-i}} \\ &= e(g,g)^{\sum_{j=1}^{t} sf(j)\prod_{\substack{i=1 \\ i\neq j}}^{t} \frac{-i}{j-i}} \\ &= e(g,g)^{sy} \\ &= e(g,Y)^s. \end{aligned}$$

Each server Γ_j sends their δ_j to Bob who then computes Δ. Lastly Bob computes $M = B \cdot \Delta^{-1}$, which is the final step of decryption.

The dynamic properties that need to be supported (in the Kim et al. interpretation) are as follows:

TTP refreshes secret key. *Same as before*

TTP adds new decryption server. *Same as before.*

TTP removes a decryption server. *Same as before.*

server Γ_i refreshes secret key s_i. *Same as before.*

In their work [7], Kim et al. successfully attacked their interpretation of the Long scheme. We find that Kim et al. misinterpreted the Long scheme [9], which we describe in Section 2.3.

2.2 Kim et al.'s Attack of Their Interpreted Long Scheme

In [7] the authors attacked their interpreted version of the Long scheme. The attack they constructed was such that it violated the decryption requirement that only t authorized servers can decrypt an encrypted message for any party. The Kim et al. attack can be summarized as follows: Suppose server Γ_w is malicious. They claim want an update of their public key P_w but rather than selecting a new secret key, suppose they wish to attack server Γ_1. They use Γ_1's public key P_1 and select $r \in \mathbb{Z}_p^*$ and compute P_1^r and sends this to the TTP claiming that P_1^r is their "new public key", calling it $P_{w,new}$. Thus $P_{w,new} = P_1^r$. The TTP not knowing that server Γ_w has misrepresented their new public key, refreshes Γ_w's shares $k_{w,new}$ and $v_{w,new}$. Here $k^{w,new} = g^{\frac{f(w)}{ID+x}} P_1^{ry}$. Because server Γ_w can compute $g^{\frac{w}{ID+x}}$ the server Γ_w now knows P_1^{ry} since the server knows r it can compute P_1^y by computing $(P_1^{ry})^{r^{-1}}$. Then using k_1 it computes $g^{\frac{f(1)}{ID+x}}$ by computing $k_1 \cdot (P_1^y)^{-1}$. Now Γ_w knows $g^{\frac{f(1)}{ID+x}}$ and $g^{\frac{f(w)}{ID+x}}$. Now together with $t-2$ other servers it can compute $g^{\frac{y}{ID+x}}$ which is $Y^{\frac{1}{ID+x}}$. Then given ciphertext C, the $t-1$ servers can compute $e(g^{S \cdot ID} \cdot X^S, Y^{\frac{1}{ID+x}}) = e(g,g)^{Sy}$, denote this by Δ. Then $M = C \cdot \Delta^{-1}$. Hence Γ_w has successfully defeated the threshold requirement since a coalition of $t-1$ servers can decrypt messages.

Note: The server Γ_w could actually complete this attack $t-1$ times and be able to decrypt by itself. Though $t-1$ refreshes may make the TTP suspicious of their behavior.

The Kim et al. attack has successfully defeated the threshold requirement. Kim argued that the only way to prevent such attack is that the system has to renew the secret shares of all decryption servers whenever one of the decryption server renews its secret key, however this is problematic in that there are n servers and m users as described in Table 1. Thus the cost is $O(mn)$, which is too much. We solved this problem constructing a more efficient scheme, we will discuss a new attack, which is relevant to all version of the Long scheme. However, we will first demonstrate that the Kim et al. interpretation of the Long scheme was incorrect.

2.3 Our Interpretation of the Long Scheme

The interpretation of the Long [9] scheme by Kim et al. [7] led to the attach described in Section 2.2, that is by their interpretation they were able to construct the attack. After observing the Long scheme [9], it became apparent to us that the error was merely in the presentation of Equation (1). That is the share k_i was expressed in the Long scheme as

$$k_i = g^{\frac{f(i)}{(ID+x)P_i^y}}. \tag{2}$$

But clearly based on the assumption that Equation (2) is correct then the δ_i (decryption subshare) is incorrect. We then observed that based on the assumption Equation (2) is correct that δ_i should be calculated as $\delta_i = e(A, k_i)^{Y^{s_i}}$ because

$$\begin{aligned} \delta_i &= e(A,k_i)^{Y^{s_i}} \\ &= e(A,g)^{Y^{s_i} \cdot \frac{f(i)}{(ID+x)P_i^y}} \\ &= e(g,g)^{S \cdot (ID+x) \cdot (P_i)^y \cdot \frac{f(i)}{(ID+x)P_i^y}} \\ &= e(g,g)^{S \cdot f(i)}, \end{aligned}$$

which is exactly what the Long scheme required. Then if a threshold of t servers (say $\Gamma_1, \ldots, \Gamma_t$) need to compute a function of the secret

$$\Delta = \prod_{j=1}^{t} \delta_j^{\prod_{\substack{i=1 \\ i \neq j}}^{t} \frac{-i}{j-i}} = e(g,Y)^s$$

The irony is that this version is not susceptible to the attack described by Kim et al. [7], and that the attempt to fix the typographical error in the Long scheme introduced the security weakness that allowed them to attack it. At the same time we note an attack on the Long scheme (both versions), as well as Kim's suggested fix.

3 Our Attack

Thus we see the Kim et al. attack was due to their interpretation of the Long scheme. Suppose Alice encrypts message M to user denoted by ID_0 then ciphertext C satisfies $C = (g^{S \cdot ID_0} \cdot X^S, e(g,Y)^S \cdot M) = (A, B)$. Now consider the following attack. Suppose a server leaves the network, in particular suppose the server is removed and is intent on causing problems to the network that removed them. We assume without loss of generality that

it is server Γ_n. Now with the removal of this server the threshold is a t out of $n-1$ and there are only $n-1$ players, all other parties are considered outsiders and outsider help does not contribute towards the threshold. For example, as an extreme, if we have t helpers who are not authentic decryption servers but for some reason posses valid shares in a t out od X scheme then they should not be able to decrypt. Suppose Γ_n gives their information $g^{\frac{f(n)}{ID+x}}$ to $t-1$ parties. The server Γ_n is now considered as an outsider and doesn't count towards the threshold, thus it should not be be considered as help to achieve a threshold that decrypt a message intended for a user. No matter which approach is taken (the original Long scheme or the Kim interpreted scheme or our interpreted scheme) all k_i and v_i are reshared out because of the dismissal of Γ_n. However if Γ_n provides the information $g^{\frac{f(n)}{ID+x}}$ to a set of $t-1$ participants, for example $\{\Gamma_1, \Gamma_2, \ldots, \Gamma_{t-1}\}$, let us call this set $\mathcal{ADV}$, then $\mathcal{ADV}$ will be able to decrypt. This is because the participants in $\mathcal{ADV}$ can use their old shares $k_{1,old}, \ldots, k_{t-1,old}$ [1]. If Γ_n sends $g^{\frac{f(n)}{ID_0+x}}$ to this group $\mathcal{ADV}$, then $\mathcal{ADV}$ can decrypt the message and $\mathcal{ADV}$ contains only $t-1$ authentic members, but this violates that no threshold less than t authentic members can decrypt. Note that any dismissed party can give their share to other members, allowing an unauthorized set to decrypt (below threshold). In fact it is possible after $t-1$ parties are dismissed that a single party could be given all shares and thus they could decrypt by themselves.

4 Our Protocol

4.1 The Protocol

Our goal is to create an efficient dynamic threshold scheme based on bilinear pairings.

Setup. Two multiplicative groups $\mathbb{G}$ and $\mathbb{G}_1$ of order prime p are selected such that there exist a bilinear map $e : \mathbb{G} \times \mathbb{G} \longrightarrow \mathbb{G}_1$. The TTP selects $g \in \mathbb{G}$ where $g \neq 0$.

Key Generation. The TTP selects a secret key denoted by y and compute the corresponding public key $Y = g^y$. The TTP selects two temporal keys x_1 and x_2 and computes $V = g^{x_2}$ and $W = g^{x_2 x_1}$. The values Y, V and W are posted on a publicly available web site, such as a bulletin board.

Each server Γ_i selects a secret key w_i and computes their public key $P_i = g^{w_i}$. They publish their public key. The TTP will keep a local copy of the server's public keys.

[1] Though the bulletin board has been updated with new shares the shareholders may have prestored the older shares.

Share Generation. The TTP selects a random polynomial $f(z) = \sum_{i=1}^{t-1} b_i \cdot z^i$ of degree $t-1$ such that $f(0) = x_2^{-1} y \mod p$, i. e. $b_0 = x_2^{-1} \cdot y$. We assume initially there are n decryption servers $\{\Gamma_1, \ldots, \Gamma_n\}$ then the TTP computes the share $k_i = f(i) \cdot P_i^y$ and $v_i = e(V, g)^{f(i)}$. The TTP publishes (k_i, v_i) on bulletin board.

User Registration of Their Identity ID. Suppose a user wishes to register their identity ID_0 with the TTP. They interact with the TTP in a communication that establishes that ID_0 is their identity (this communication to achieve this is outside our scope). Once this is established the TTP publishes Z_{ID_0} where $Z_{ID_0} = g^{\frac{1}{ID_0+x_1}}$ onto the public site (bulletin board).

Encryption of Message M to User with Identity ID_0. Suppose *Alice* would like to transmit message M to *Bob* privately. She gets *Bob*'s identification, denoted by ID_0, as well as the TTP's public key Y and the two temporal key X. She then encrypts message M with public key ID_0, by picking up a random $S \in \mathbb{Z}_p^*$ and computes the ciphertext C by $C = (W^S \cdot V^{S \cdot ID_0}, e(g, Y)^S \cdot M) = (A, B)$.

Generation of Decryption Server Γ_i Decryption Shares. User Bob with identity ID_0 requests to the decryption servers that the ciphertext $C = (A, B)$ be decrypted. Assuming t servers respond, say $\{\Gamma_{i_1}, \ldots, \Gamma_{i_t}\}$, each of these servers will compute a decryption share based on the ciphertext and their share of the decryption key. For each i_r, server Γ_{i_r} computes the decryption share δ_{i_r} where δ_{i_r} satisfies

$$\begin{aligned} \delta_{i_r} &= e(A^{k_{i_r} \cdot Y^{-w_{i_r}}}, Z_{ID_0}) \\ &= e(A^{f(i_r)}, Z_{ID_0}) \\ &= e(g^{(x_2 x_1 S + x_2 S I D_0) f(i_r)}, g^{\frac{1}{ID_0+x_1}}) \\ &= e(g, g)^{S x_2 \cdot f(i_r)}. \end{aligned}$$

Here w_{i_r} is the server Γ_{i_r}'s secret key.

Decryption. The combiner using the decryption shares $\delta_{i_1}, \ldots, \delta_{i_t}$ from $\Gamma_{i_1}, \ldots, \Gamma_{i_t}$, respectively computes Δ by

$$\begin{aligned} \Delta &= \prod_{r=1}^{t} \delta_{i_r}^{\prod_{v=1, v\neq r}^{t} \frac{-i_v}{i_r - i_v}} \\ &= \prod_{r=1}^{t} (e(g, g)^{S x_2 \cdot f(i_r)})^{\prod_{v=1, v\neq r}^{t} \frac{-i_v}{i_r - i_v}} \\ &= e(g, g)^{\sum_{r=1}^{t} S x_2 \cdot f(i_r) \prod_{v=1, v\neq r}^{t} \frac{-i_v}{i_r - i_v}} \\ &= e(g, g)^{x_2 x_2^{-1} y S} \\ &= e(g, g)^{yS}. \end{aligned} \quad (3)$$

The message M can be computed by $M = B \cdot \Delta^{-1}$.

Verifiability.

1) Each user ID can verify their public key by computing $e(V^{ID}W, Z_{ID})$ and comparing it with $e(g, V)$.
2) Each server Γ_i verifies k_i by computing $e(V, g)^{k_i \cdot Y^{-w_i}}$ and comparing it to $v_i = e(V, g)^{f(i)}$.
3) Any server Γ_i can select t values $v_{j_1}, \ldots, v_{j_t}$ and compute

$$\begin{aligned} & \prod_{r=1}^{t} v_{i_r}^{\prod_{w=1, w\neq r}^{t} \frac{-i_w}{i_r - i_w}} \\ = & \prod_{r=1}^{t} (e(V, g)^{f(i_r)})^{\prod_{w=1, w\neq r}^{t} \frac{-i_w}{i_r - i_w}} \\ = & e(V, g)^{\sum_{r=1}^{t} f(i_r) \prod_{w=1, w\neq r}^{t} \frac{-i_w}{i_r - i_w}} \\ = & e(g, g)^{y} \\ = & e(g, Y). \end{aligned}$$

TTP Adds New Decryption Server. Assume that the TTP needs to add server Γ_{n+1}. The TTP computes $k_{n+1} = f(n+1) \cdot P_{n+1}^{y}$ and $v_{n+1} = e(V, g)^{f(n+1)}$.

TTP Removes a Decryption Server. Without loss of generality suppose the TTP needs to remove (or deactivate) server Γ_n, thus producing a t out of $n-1$ threshold decryption service [2]. First the TTP selects a new $x_{2,new} \in \mathbb{Z}_p^*$ and computes $V_{new} = g^{x_{2,new}}$ and $W_{new} = g^{x_1 x_{2,new}}$. Then for $i = 1, \ldots, n-1$ the TTP computes $k_{i,new} = f(i) \cdot P_i^{y}$ and $v_{i,new} = e(V, g)^{f(i)}$.

Server Γ_i Refreshes Their Secret Key. Suppose server Γ_i contacts the TTP and notifies them they wish to refresh the secret key. The server sends the TTP $P_{i,new}$. The TTP then selects a $R \in \mathbb{Z}_p^*$ and sends the challenge g^R to Γ_i. The server sends $g^{R \cdot DLOG(P_{i,new})}$ to the TTP. Here $DLOG(P_{i,new})$ is a u such that $g^u = P_{i,new}$. The TTP compares $e(g, g^{R \cdot DLOG(P_{i,new})})$ to $e(g^R, P_{i,new})$. If they are equal then the TTP updates k_i. Otherwise the sever Γ_i has lied and the TTP may punish (even remove the server).

TTP Removes a User with ID_0. Suppose that the TTP must dismiss user with identity ID_0. The TTP selects $x_{1,new} \in \mathbb{Z}_p^*$ and computes $W_{new} = g^{x_2 x_{1,new}}$ The TTP removes $g^{\frac{1}{ID_0 + x_1}}$ from the bulletin board. For all users ID with $ID \neq ID_0$ the TTP computes $g^{\frac{1}{ID + x_{1,new}}}$ and places it on the bulletin board.

4.2 Security Analysis

We assume the following security assumptions: Both the DLOG and CDH problems are hard in the presence in of a bilinear map. We assume that a threshold many servers act correctly. That is, if we have a t out of n threshold scheme then any t or many serves act correctly. If t or more servers are malicious then since they possess the threshold we assume that their actions are correct. Once a threshold is reached we cannot claim protection.

Theorem 1. *Given a coalition of less than t active servers, then the coalition cannot decrypt any validly constructed (using current public values) ciphertext C.*

Proof. Let $\rho < t$ and let $\Gamma_1, \Gamma, \ldots, \Gamma_\rho$ denote a set of ρ many active servers, a coalition which attempts to decrypt the server. Because shares have been distributed in a t out of n manner the coalition cannot decrypt the ciphertext without additional information beyond the shares distributed to the ρ servers. This additional information must come from servers who are no longer active (due to the threshold requirement). Let $\Phi_1, \ldots, \Phi_\omega$ denote deactivated servers who contribute (possibly actively or passively) with the coalition $\Gamma_1, \Gamma, \ldots, \Gamma_\rho$. Then $\omega + \rho \geq t$. Recall ciphertext $C = (W_{curr}^{S} \cdot V_{curr}^{S \cdot ID_0}, e(g, Y)^S \cdot M) = (A, B)$ where S is random, $W_{curr} = g^{x_1 x_{2,curr}}$ and $V_{curr} = g^{x_{2,curr}}$.

As $\Gamma_1, \Gamma, \ldots, \Gamma_\rho$ are active servers, they have shares $k_{\Gamma_j, curr}$ and $v_{\Gamma_j, curr}$ constructed for use with W_{curr} and V_{curr}. Now $\Phi_1, \ldots, \Phi_\omega$ are deactivated servers, they may possess "dated shares" (perhaps downloading from bulletin board earlier). They possess shares $k_{\Phi_i, time_i}$ and $v_{\Phi_i, time_i}$ constructed for use at time $time_i$ where W_{time_i} and V_{time_i}, note that $time_i < curr$. However the shares $k_{\Phi_i, time_i}$ and $v_{\Phi_i, time_i}$ do not work with shares $k_{\Gamma_j, curr}$ and $v_{\Gamma_j, curr}$ because $W_{curr} \neq W_{time_i}$ and $V_{curr} \neq V_{time_i}$ for all i. Therefore the only alternative is that at least t members from $\{\Gamma_1, \Gamma, \ldots, \Gamma_\rho, \Phi_1, \ldots, \Phi_\omega\}$ share some time $time_0$ such that each of these t members possess $k_{\Gamma_j, time_0}$ and $v_{\Gamma_j, time_0}$ or $k_{\Phi_i, time_0}$ and $v_{\Phi_i, time_0}$, respectively. Then for server i', this server i' will be able to compute $e(g, g)^{S x_{2,time_0} f_{time_0}(i')}$ where the constant coefficient of $f_{time_0}(x)$ is $x_{2,time_0}^{-1} y$. Now when all t servers apply their $\delta_{i'}$ into Equation (3), the corresponding Δ satisfies $\Delta = e(g, g)^{x_{2,curr} x_{2,time_0}^{-1} yS} \neq e(g, g)^{yS}$.

Therefore ρ many active servers, with $\rho < t$, cannot decrypt the ciphertext. □

Theorem 2. *If active server Γ_i refreshes their public key $P_{i,curr}$ then Γ_i knows the discrete log of $P_{i,curr}$.*

Proof. This follows directly from the refresh key protocol and the fact that the DLOG problem is "hard". □

4.3 Efficiency of Our Schemes

The the cost of executing the refresh properties are (in big O notation) is described in Table 2.

Here in Table 2, the value m represents the number of users and n represents the number of servers. In most applications one should expect that m is significantly larger

[2] We characterize the $n-1$ servers as active servers.

Table 2: Computational cost

operation	computational cost of our scheme	computational cost of the Long scheme
TTP refreshes secret key	$O(n)$	$O(mn)$
TTP adds new decryption server	$O(1)$	$O(m)$
TTP removes a decryption server	$O(n)$	$O(mn)$
server Γ_i refreshes their secret key	$O(1)$	$O(m)$
TTP removes a user with ID_0	$O(m)$	not discussed

than n. The cost is only for computational purposes there will also be a communication cost, although one may expect the communication cost between the TTP and the bulletin board is significantly less than the communication cost between the TTP and a server.

The above table demonstrates that our scheme is significantly more efficient than the existing schemes. Furthermore we have added a new service the dismissal of a user.

5 Conclusion

In this paper we have discussed dynamic threshold decryption scheme using the bilinear pairing. We have analyzed previous scheme noting their weaknesses, in particular their inefficiency. We have constructed a new scheme that is significantly more efficient than the previous schemes.

These schemes all rely on the use of a bulletin board to achieve the necessary dynamic properties. It remains an open problem if a dynamic public key scheme can be constructed without the use of a bulletin board. It remains an open problem if one can construct a dynamic scheme which uses bilinear pairing that allows the dismissal of users in $O(1)$ computations.

References

[1] J. Baek, J. Newmarch, R. S. Naini, and W. Susilo, "A survey of identity-based cryptography," in *Proceedings of Australian Unix Users Group Annual Conference*, pp. 95–102, 2004.

[2] G. R. Blakley, "Safeguarding cryptographic keys," in *Proceedings of AFIPS '79*, vol. 48, pp. 313–317, 1979.

[3] D. Boneh and M. Franklin, "Identity-based encryption from the weil pairing," in *Advances in Cryptology (Crypto'01)*, LNCS 2139, pp. 213–229, Springer-Verlag, 2001.

[4] D. Boneh, B. Lynn, and H. Shacham, "Short signatures from the weil pairing," *Journal of Cryptology*, vol. 17, no. 4, pp. 297–319, 2004.

[5] L. Chen, D. Gollmann, C. Mitchell, and P. Wild, "Secret sharing with reusable polynomials," in *Proceedings of the Second Australasian Conference on Information Security and Privacy (ACISP'97)*, pp. 183–193, Springer-Verlag, 1997.

[6] Y. Desmedt, "Society and group oriented cryptography: A new concept," in *Advances in Cryptology (Crypto'87)*, pp. 120–127, 1987.

[7] K. Kim, S. Lim, I. Yie, and K. Kim, "Cryptanalysis of a dynamic threshold decryption scheme," *Communications of the Korean Mathematical Society*, vol. 24, no. 1, pp. 153–159, 2009.

[8] C. S. Laih, L. Ham, J. Y. Lee, and T. Hwang, "Dynamic threshold scheme based on the definition of cross-product in an n-dimensional linear space," *Journal Information Science and Engineering*, vol. 7, pp. 13–23, 1991.

[9] Y. Long and K. F. Chen, "Construction of dynamic threshold decryption scheme from pairing," *International Journal of Network Security*, vol. 2, no. 2, pp. 111–113, 2006.

[10] A. Shamir, "How to share a secret," in *Communications of the ACM*, vol. 22, pp. 612–613, 1979.

[11] A. Shamir, "Identity based cryptosystems and signature schemes," in *Advances in Cryptology (Crypto'84)*, LNCS 196, pp. 47–53, Springer-Verlag, 1984.

[12] H. M. Sun and S. P. Shieh, "Construction of dynamic threshold schemes," *Electronics Letters*, vol. 30, pp. 2023–2025, 1994.

Brian King received a Ph.D. in mathematics (1990) and a Ph.D. in Computer Science (2000). He is currently an associate professor of Electrical and Computer Engineering at Indiana Univ. Purdue Univ. Indianapolis (IUPUI). Prior to joining IUPUI he worked in the Security Technologies Lab at Motorola Research Labs. His research interests include: wireless security, cryptography, threshold cryptography and low-complexity cryptosystems.

Analysis and Improvement of Patient Self-controllable Multi-level Privacy-preserving Cooperative Authentication Scheme

Yang Zhao[1], Feng Yue[1], Songyang Wu[2], Hu Xiong[1,3], and Zhiguang Qin[1]
(Corresponding author: Songyang Wu)

School of Computer Science and Engineering & University of Electronic Science and Technology of China[1]
No. 4, North Jianshe Road, Chenghua District, chengdu, Sichuan, 610054, China
The Third Research Institute of Ministry of Public Security[2]
No. 76, Yueyang Road, Shanghai, 201204, China
(Email: wusongyang@stars.org.cn)
State Key Laboratory of Information Security, Institute of Software & Chinese Academy of Sciences[3]
No. 19, Yuquan Road, Shijingshan District, Beijing, 100190, China
(Received Mar. 29, 2015; revised and accepted May 15 & June 4, 2015)

Abstract

In 2014, a patient self-controllable multi-level privacy-preserving cooperative authentication scheme (PSMPA) was proposed for attempting to address the issue of data confidentiality and patients' identity privacy simultaneously when the personal healthcare record (PHR) is shared in the distributed m-healthcare cloud computing system. In this paper, we show the PSMPA scheme fails to achieve the two goals under the collusion attack. Furthermore, the scheme also suffers from forgery attack because of a flawed design in the transcript simulation phase. In order to avoid the attacks, we propose an improved PHR sharing scheme by incorporating ciphertext policy attribute-based encryption (CP-ABE) and attribute-based signature (ABS) as a possible solution.

Keywords: Access control, attribute-based encryption, data confidentiality, identity privacy

1 Introduction

Motivated by the remarkable development of cloud computing, more and more significant data is stored into the cloud for sharing, including personal health record (PHR) absolutely. The e-healthcare service attracts much more attention than the traditional approaches due to its fascinating features such as high efficiency, universal accessibility and low cost. The patients can share their PHRs in the cloud to obtain treatment recommendations from physicians or to provide medical research institutions with precious medical information. However, on account of storing PHRs in the cloud far away from the patients, the PHRs are out of their physical control. The data confidentiality and patient's identity privacy will face enormous threatens which are bound to the obstacles of its wide adoption. To minimize users' concerns as far as possible, a lot of data sharing schemes in distributed cloud computing system have been proposed so far where cryptography is utilized popularly.

It's natural to think of leveraging the access control in the e-healthcare scheme. Access control enables the patients to delegate different privilege for accessing the PHRs to whoever they desire with freedom. ABE is considered as the most optimal solution to realizing fine-grained access control for sensitive data in the cloud environment. A number of literatures on ABE have been published in the past. Especially, in 2006 Goyal *et al.* [3] proposed key-policy attribute-based encryption fine-grained access control of encrypted data which makes key management more efficient during data sharing. Similarly, Bethencourt *et al.* [1] put forward the concept of ciphertext-policy attribute-based encryption which is parallel with KP-ABE. CP-ABE and KP-ABE are applied to different scenes dependent on their respective specialties. Nevertheless, both of them are short of efficient and dynamic attribute revocation mechanism which is indispensable. Based on ABE mentioned in [1, 3], [4, 11, 12] are proposed one after another. However, the single attribute authority that is responsible for distributing attributes becomes the bottleneck of these schemes inevitably. In 2009, Chase *et al.* [2] figured out a solution called multiple-authority ABE where multiple attribute authorities are requested to involve in distributing attributes. On the basis of [2], Li *et al.* [6, 7] divided the members in the cloud into various security domains for the purpose of decreasing the key management complexity

further. In 2013, Lee *et al.* [5] carried on a comprehensive survey on the existing ABE schemes and ran an extended analysis on KP-ABE and CP-ABE. In the same year, Li *et al.* [8] proposed the first multi-authority attribute based encryption scheme realizing such expressive access policy and constant ciphertext size. As in the real world circumstance, the attributes are always in the different levels, Liu *et al.* [9] proposed a scheme called ciphertext-policy hierarchical attribute based encryption in 2014. The above schemes mainly concentrate on achieving data confidentiality, while the user's identity privacy is neglected.

Recently, Zhou *et al.* [13] proposed a novel PSMPA aiming at guaranteeing the patient's identity privacy. The PHRs are divided into patient's identity information and healthcare data creatively and each of them is encrypted respectively. No one can decrypt the patient's authentic identity except the directly authorized physicians the patient has appointed personally. By this means, they claimed their scheme can satisfy the requirement of identity privacy. However, we find that the parent's identity privacy and healthcare data are vulnerable because the PSMPA is unable to resist the collusion attack from the dishonest physicians. In addition, the scheme also suffers from forgery attack because of a flawed design in the transcript simulation phase. Incorporating CP-ABE and ABS, we propose an improved PHRs sharing scheme as a possible solution.

The rest paper is organized as follows: In the Section 2, we review Zhou *et al.* [13]'s PSMPA scheme in detail. Our attacks against the PSMPA scheme are demonstrated in Section 3. In Section 4, we show a possible solution and Section 5 is the final conclusion.

2 Review of the PSMPA Scheme

In this section, we carry out a detailed statement on the PSMPA scheme to prepare for the analysis and the attacks in Section 3.

2.1 Network Model

As is illustrated in Figure 1, in the m-healthcare cloud computing system, all the members are classified into three levels of security: the directly authorized physicians such as Bob in the local healthcare provider, the indirectly authorized physicians such as Jack, Tom and Jim in the remote healthcare providers and the unauthorized persons such as Black. The directly authorized physicians are authorized by the patients and can not only access the patient's personal health record but also recognize the patient's identity. The indirectly authorized physicians are authorized by the directly authorized physicians for medical consultant or some research purposes (since they are not authorized by the patients, we call them 'indirectly authorized' instead). The only right they have is accessing the personal health record, but not the patient's identity. For the unauthorized persons, neither could be obtained.

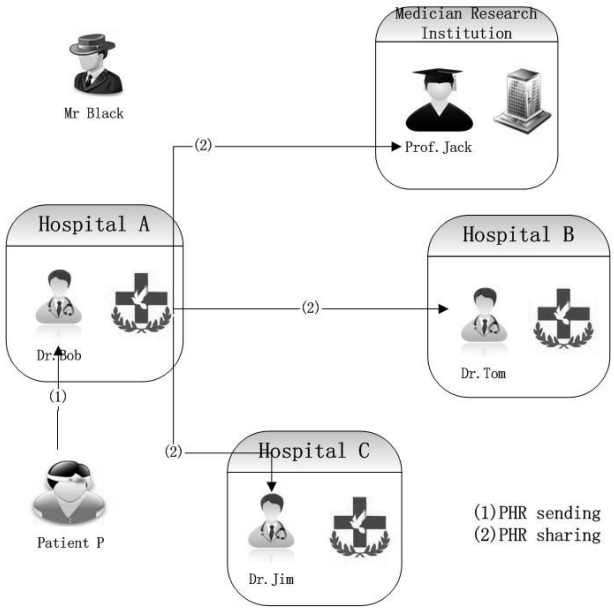

Figure 1: An overview of the M-healthcare cloud computing system

2.2 Authorized Accessible Privacy Model (AAPM)

A novel attribute based designated verifier signature scheme (ADVS) is proposed by Zhou *et al.* [13] to realize three levels of security and privacy requirement in distributed m-healthcare cloud computing system which is mainly constituted of the following five algorithms: Setup, Key Extraction, Sign, Verify and Transcript Simulation Generation. Suppose the universe set of attributes is U. If and only if $\mathbb{A}(\omega) = 1$ where ω is selected from U, We say an attribute set ω satisfies a specific access structure $\mathbb{A}$. The five phases are presented as follows.

Setup. The algorithm takes 1^l as input, where l is the security parameter. It outputs public parameters and y as the master key for the central attribute authority.

Key Extract. Assume that a physician requests the attribute keys for an attribute set $\omega_D \in U$. If he is qualified to be issued with sk_D for these attributes, the attribute authority produces sk_D for him.

Sign. The patient takes as input his private key sk_P, the uniform public key pk_D of the healthcare provider which the physicians work in and a personal healthcare information m to generate a signature σ. Namely, $\sigma \leftarrow Sign(sk_P, pk_D, m)$.

Verify. Suppose that a physician wants to validate the correction of a signature σ which contains an access structure $\mathbb{A}$ and owns a subset of attributes $\omega_J \subseteq \omega_D$ satisfying $\mathbb{A}(\omega_J) = 1$, a deterministic verification algorithm can be executed. Once receiving a signature σ, he uses his attribute private key sk_D and the patient's public key pk_P, then outputs the message m

and $True$ if the signature is correct, or $\perp$ otherwise. Namely, $\{True, \perp\} \leftarrow Verify(sk_D, pk_P, m, \sigma)$.

Transcript Simulation Generation. Through the Transcript Simulation algorithm, the directly authorized physicians who kept the authorized private key sk_D can always produce identical distributed transcripts indistinguishable from the signature which is received from the patient.

2.3 PSMPA Design

In this section, we introduce the proposed PSMPA to implement AAPM mentioned above, realizing three different levels of security and privacy requirements. Most of the notations which are useful in our scheme are showed in Table 1 with the corresponding description.

Setup. Assume that $\mathbb{G}_0$ and $\mathbb{G}_1$ are two bilinear groups of prime order p and g is a generator of $\mathbb{G}_0$. Moreover, let $\hat{e} : \mathbb{G}_0 \times \mathbb{G}_0 \to \mathbb{G}_1$ denote a bilinear map. Pick $g_1 \in \mathbb{G}_0$, $y \in \mathbb{Z}_p^*$ at random and compute $g_2 = g^y$. We additionally employ three collision-resistant hash functions: $H_0 : \{0,1\}^* \to \mathbb{G}_0$, $H_1 : \{0,1\}^* \to \mathbb{Z}_p^*$ and $H_2 : \mathbb{G}_1 \to \{0,1\}^{k_{Enc}}$ where k_{Enc} is the length of symmetric key in the secure private key encryption construction chosen by the patient. Then, define the attributes in universe U as elements in $\mathbb{Z}_p$. If $q_x(\cdot)$ is a polynomial related to leaf nodes, a default attribute set from $\mathbb{Z}_p$ with the size of $d_x - 1$ is denoted by $\Psi_x = \{\Psi_1, \Psi_2, \cdots, \Psi_{d_x-1}\}$ in the access tree.

Key Extract. The patient choose $b \in \mathbb{Z}_p^*$ and $B = g^b$ as his private key and public key. We define the patient's registered local healthcare provider's uniform private key is $sk^{HP} = hc$ and the corresponding public key is $pk^{HP} = g_1^{hc}$. Both of the keys are shared by each physician working in it. The attribute private key of the physician can be

$$sk_D = (\gamma_i, \delta_i) = ((g_1 H_0(i))^{q_x(i)}, g^{q_x(i)})_{i \in \omega_D \cup \Psi_x},$$

and the public parameters are

$$(p, g, \hat{e}, \mathbb{G}_0, \mathbb{G}_1, H_0, H_1, H_2, g_1, g_2).$$

Sign. The signing algorithm produces a signature of the patient's personal health information m which can only be decoded and validated by the directly authorized physicians whose sets of attributes enable to satisfy the patients' requirements. First of all, the patient need to construct a polynomial $q_x(\cdot)$ for each node x in Γ of the degree $D_x = d_x - 1$.

Beginning with the root node R, the algorithm chooses a random $y \in \mathbb{Z}_p$ and sets $q_R(0) = y$. Then, it chooses $d_R - 1$ other points on the polynomial q_R randomly to define it completely. For any other node x, it sets $q_x(0) = q_{parent(x)}(index(x))$ and chooses $d_x - 1$ other points randomly to completely define $q_x(\cdot)$.

To sign a message m with the verification predicate Γ, for the leaf node x in the access tree Γ, let the current threshold required for the physician be k_x. For the leaf node polynomial $q_x(\cdot)$, the patient randomly selects a default subset $\Psi'_x \subseteq \Psi_x$ with $|\Psi'_x| = d_x - k_x$ and calculates $B_{P_i} = H_0(i)^b$ for $i \in \omega_x^* \cup \Psi'_x$. Then, he can derive the corresponding keys for authentication

$$\begin{aligned} K_{Encp} &= \hat{e}(g_1, g_2)^b, \\ K_{Enc} &= H_2(K_{Encp}), \\ K_{Sig} &= K_{Encp}\hat{e}(pk^{HP}, g_2). \end{aligned}$$

Finally, the patient randomly selects $r_i \in \mathbb{Z}_p^*$ for each $i \in \omega_x^* \cup \Psi'_x$, publishes $g^{r_i} (i \in \omega_x^* \cup \Psi'_x)$ and completes the signature as follows

$$\begin{aligned} \sigma' &= H_1(m \parallel K_{Sig}), \\ C_0 &= E_{pk^{HP}}(B \parallel B_{P_i}), \\ C &= E_{K_{Enc}}(m), \\ \sigma''_i &= \{H_0(i)^{r_i}\}_{i \in \omega_x \cup \Psi'_x}, \\ \sigma''' &= H_0(m)^b, \end{aligned}$$

where $E_{pk^{HP}}(\cdot)$, $E_{K_{Enc}}(\cdot)$ are secure public key and private key encryptions chosen by the patient. At last, he can export the signature $\sigma = (\omega_x^*, C_0, C, \sigma', \sigma''_i, \sigma''')$.

Verify. Upon obtaining the signature σ, the physicians working in the patient's registered local healthcare provider can firstly decipher $B \parallel B_{P_i} = D_{sk^{HP}}(C_0)$, where $D_{sk^{HP}}(\cdot)$ is the decryption algorithm of the public key encryption. If the set of attributes kept by the physician satisfies the access tree Γ, the patient is able to further finish the verification by implementing a recursive algorithm illustrated as follows.

For the leaf node x, to testify the signature with the node predicate, that is to prove possessing at least k_x attributes among an attribute set ω_x with the size of n_x, the physician firstly selects a subset $\omega_J \subseteq \omega_D \cap \omega_x^*$ of the size k_x, chooses $r'_i \in_R \mathbb{Z}_p^*$ for each $i \in \omega_x^* \cup \Psi'_x$ and computes

$$\begin{aligned} V' &= \prod_{i \in \omega_J \cup \Psi'_x} \gamma_i^{\Delta_{i,\omega_J \cup \Psi'_x}(0)}, \\ V'' &= \prod_{i \in \omega_x^* \cup \Psi'_x} (\sigma''_i)^{r'_i}, \\ V''' &= \prod_{i \in \omega_J \cup \Psi'_x} \hat{e}(B_{P_i}, \delta_i^{\Delta_{i,\omega_J \cup \Psi'_x}(0)} g^{r_i r'_i}), \quad (1) \\ V'''' &= \prod_{i \in \omega_x^* \setminus \omega_J} \hat{e}(B_{P_i}, g^{r_i r'_i}), \quad (2) \end{aligned}$$

Table 1: Notations in the PSMPA

Notation	Description
d_x	Threshold for node x in access tree Γ
k_x	Number of attributes required to be owned by the patient w.r.t. node x
$q_x(\cdot)$	$D_x = d_x - 1$ -degree polynomial assigned to node x
Ψ_x	A default attribute set of size $d_x - 1$ for node x
sk^{HP}	Uniform private key of the healthcare center
pk^{HP}	Uniform public key of the healthcare center
ω_D	The set of attributes owned by the physician
sk_D	Private key of the physician
ω_x^*	Attributes in predicate of node x for physicians
Ψ'_x	A subset of default attribute set of size $d_x - k_x$ chosen by the patient
K_{Enc}/K_{Dec}	Symmetric key for message encryption/decryption
K_{Sig}	Signing key for ADVS
ω_J	The subset of physician's attribute set of size k_x chosen to satisfy the predicate
H_0,H_1,H_2	Hash functions mapping $\{0,1\}^* \to \mathbb{G}_0$, $\{0,1\}^* \to \mathbb{Z}_p^*$ and $\mathbb{G}_1 \to \{0,1\}^{k_{Enc}}$

and

$$\begin{aligned} K^x_{Decp} &= \frac{\hat{e}(V'V'', B)}{V'''V''''} \\ &= \frac{\hat{e}(g_1^{q_x}(0)\prod_{i\in\omega_J\cup\Psi'_x} H_0(i)^{q_x(i)\Delta_{i,\omega_J\cup\Psi'_x}(0)+r_i r'_i}, g^b)}{\prod_{i\in\omega_J\cup\Psi'_x}\hat{e}(H_0(i)^b, g^{q_x(i)\Delta_{i,\omega_J\cup\Psi'_x}(0)+r_i r'_i})} \\ &\quad \cdot\frac{\hat{e}(\prod_{i\in\omega^*_x\setminus\omega_J} H_0(i)^{r_i r'_i}, g^b)}{\prod_{i\in\omega^*_x\setminus\omega_J}\hat{e}(H_0(i)^b, g^{r_i r'_i})} \\ &= \hat{e}(g_1^{q_x(0)}, g^b). \end{aligned} \quad (3)$$

We now consider the recursive case when x is a non-leaf node. The verification algorithm will proceed as follows. For all nodes z that are children of x, it calls the same verification algorithm with respect to itself and stores the corresponding partial output as F_z. Let $\mathbb{S}_x$ be an arbitrary k_x -sized set of child nodes z such that $F_z \neq \perp$. If no such set exists, the node will not be satisfied and the function will return $\perp$. Then, the physicians can compute

$$\begin{aligned} K^x_{Decp} &= \hat{e}(F_x, B) = \hat{e}(\prod_{z\in\mathbb{S}_x} F_z^{\Delta_{i,\mathbb{S}'_x}(0)}, g^b) \\ &\quad (i = index(x)\ and\ \mathbb{S}'_x = \{index(z)\} : z \in \mathbb{S}_x) \\ &= \hat{e}(\prod_{z\in\mathbb{S}_x} g_1^{q_z(0)\Delta_{i,\mathbb{S}'_x}(0)}, g^b) \\ &= \hat{e}(\prod_{z\in\mathbb{S}_x} g_1^{q_{parent(z)}(index(z))\Delta_{i,\mathbb{S}'_x}(0)}, g^b) \\ &= \hat{e}(\prod_{z\in\mathbb{S}_x} g_1^{q_x(i)\Delta_{i,\mathbb{S}'_x}(0)}, g^b) = \hat{e}(g_1^{q_x(0)}, g^b). \end{aligned}$$

Until now, we have defined the verification function for each node in the access tree Γ. By utilizing the recursive algorithm defined above, the physicians can complete verification by simply calling the function on the root node R of the access tree Γ. Finally, the directly authorized physician computes

$$\begin{aligned} K_{Decp} &= \hat{e}(F_R, B) = \hat{e}(g_1^{q_R(0)}, g^b) \\ &= \hat{e}(g_1, g_2)^b, \quad (4) \\ K_{Dec} &= H_2(K_{Decp}), m = D_{K_{Dec}}(C), \quad (5) \end{aligned}$$

and verifies whether both

$$\begin{aligned} \hat{e}(g, \sigma''') &= \hat{e}(B, H_0(m)), \quad (6) \\ H_1(m \parallel K_{Decp}\hat{e}(g_1, g_2)^{hc}) &= H_1(m \parallel K_{Sig}) \\ &= \sigma', \quad (7) \end{aligned}$$

hold, where $D_{K_{Dec}}(\cdot)$ is the decryption algorithm for the private key encryption. If Equations (6) and (7) hold simultaneously, the physician outputs $True$; otherwise, outputs $\perp$.

Transcript Simulation. Once receiving the medical consultation or research, the directly authorized physician creates a protected session secret SS_j which is unique to each consultation j made for each patient. Next, he can output the transcript simulation σ_T which is broadcasted to indirectly authorized physicians by operating the following procedures. Firstly, he computes $K^T_{Decp} = K^{H_1(SS_j)}_{Decp}$, $K^T_{Dec} = H_2(K^T_{Decp})$ to encrypt a specific message m to C_T and computes $\sigma'_T = H_1(m \parallel K^T_{Decp}\hat{e}(pk^{HP'}, g_2)) = H_1(m \parallel K^T_{Sig})$, in which $pk^{HP'}$ is the public key of the hospital which the indirectly authorized physician works in. After that, he can compute $B_T = B^{H_1(SS_j)}$, $B^T_{P_i} = B^{H_1(SS_j)}_{P_i}$ and encrypt them as $C^T_0 = E_{pk^{HP'}}(B_T \parallel B^T_{P_i})$. In the end, he calculates $\sigma'''_T = (\sigma''')^{H_0(SS_j)}$ and generates the transcript simulation as $\sigma_T = (\omega^*_x, C^T_0, C_T, \sigma'_T, \sigma''_i, \sigma'''_T)$ which is indistinguishable from the original signature σ for the indirectly authorized physician.

3 Attacks Against the PSMPA Scheme

Through analysis and discussion, two primary flaws can be found in the patient self-controllable multi-level privacy-preserving cooperative authentication scheme (PSMPA).

3.1 Collusion Attack

The indirectly authorized physicians who satisfy the attributes requirement are capable of colluding with the directly authorized physicians who work in the patient's registered hospital and don't satisfy the attribute requirements.

In the proposed scheme, the above two kinds of members can exchange information and get what they shouldn't have the right to get. The former can recognize the patient's authentic identity and the latter can obtain the healthcare data. The analysis and collusion attack are presented as follows.

In the sign phase, the authentic identity of the patient B is encrypted by the public key of his registered local healthcare provider just as $C_0 = E_{pk^{HP}}(B \parallel B_{P_i})$. Since all of the physicians working in the patient's registered hospital possess the private key sk^{HP} of the healthcare provider which they work in, they are able to decipher the authentic identity B by computing $B \parallel B_{P_i} = D_{sk^{HP}}(C_0)$, no matter whether they satisfy the access tree or not.

After the decryption, the directly authorized physicians can share $B \parallel B_{P_i}$ and the corresponding signature $\sigma = (\omega_x^*, C_0, C, \sigma', \sigma_i'', \sigma''')$ with the dishonest indirectly authorized physicians who satisfy the access tree. With their own attribute private key $sk_D = (\gamma_i, \delta_i)$ and the value of $B \parallel B_{P_i}$, the indirectly authorized physicians enable to calculate the value of V''' and V'''' just like Equations (1) and (2) in the PSMPA, when x represents a leaf node in the access tree. Then they also can compute Equation (3).

Finally, the indirectly authorized physicians implement the same operations as the primitive paper to compute K_{Decp}, K_{Dec} and the healthcare information m easily by computing Equations (4) and (5).

In this way, the indirectly authorized physicians are able to share the healthcare data m with the directly authorized physicians, and they also get the authentic identity B illegitimately from the latter. The collusion attack succeeds after the cooperation.

3.2 Forgery Attack

In the transcript simulation phase, because of the flawed design, the directly authorized physicians are equipped with the ability of deceiving the indirectly authorized ones through sharing fake healthcare data, while the latter do not notice that.

The public key infrastructure (PKI) is utilized to issue the certificate for user's public key in the paper. The PKI requires that if a patient wants to get a public key certificate from certificate authority (CA), he must pass the identity verification. In the transcript simulation phase, the directly authorized physicians randomize the patient's authentic identity by an exponent arithmetic $B^{H_1(SS_j)}$ in order to protect the patient's privacy. The blinded identity is certain to fail to get the corresponding certificate from CA. Now that the patient's public/private key pair $B_T = B^{H_1(SS_j)}$ and $b_T = b(H_1(SS_j))$ is fake completely, the directly authorized physician enables to simulate a forged signature for any healthcare data m^* he likes with a fake identity B^* and cheat the indirectly authorized physician as follows.

1) Signature Generation:

   a. The dishonest directly authorized physician randomly selects $b^* \in \mathbb{Z}_p^*$ as a nonexistent patient's private key and computes the corresponding public key $B^* = g^{b^*}$.

   b. A suit of new secret values will be produced with the help of the fake private and public key pair B^*/b^* through computing

$$\begin{aligned} K_{Encp}^* &= \hat{e}(g_1, g_2)^{b^*}, \\ K_{Enc}^* &= H_2(K_{Encp}^*), \\ K_{Sig}^* &= K_{Encp}^* \hat{e}(pk^{HP'}, g_2), \end{aligned}$$

   where HP' denotes the public key of the healthcare provider which the indirectly authorized physician works in.

   c. The forged signature can be computed as follows.

$$\begin{aligned} \sigma'^* &= H_1(m^* \parallel K_{Sig}^*), \\ C_0^* &= E_{pk^{HP'}}(B^* \parallel B_{P_i}^*), \\ C^* &= E_{K_{Enc}^*}(m^*), \\ \sigma_i''^* &= \{H_0(i)^{r_i}\}_{i \in \omega_x^* \cup \Psi_x'}, \\ \sigma'''^* &= H_0(m^*)^{b^*}, \end{aligned}$$

   where $B_{p_i}^* = H_0(i)^{b^*}$ for each $i \in \omega_x^* \cup \Psi_x'$. Then, the forged signature will be $\sigma^* = (\omega_x^*, C_0^*, C^*, \sigma'^*, \sigma_i''^*, \sigma'''^*)$.

2) Signature Verification:

   a. After receiving the signature σ^*, the indirect authorized physician firstly utilizes the healthcare provider's secret key $sk^{HP'}$ to decipher the patient's identity information by calculating $B^* \parallel B_{P_i}^* = D_{sk^{HP'}}(C_0^*)$.

   b. The indirectly authorized physician will be able to decipher the healthcare information m^* and verify the correction of the signature according to the other procedures in the verification phase of PSMPA.

Through the verification, the indirectly authorized physician is convinced that m^* is the healthcare information he desires and B^* is the corresponding patient's authentic identity without being aware of being cheated.

In this case, the correction of transcript simulation is entirely dependent on the honesty of the directly authorized physicians. Unfortunately, the probability of this honesty guarantee is negligible in practice such that this type of data sharing mechanism is unrealistic.

4 Possible Solution

In this section, we provide a possible solution to avoid the above two attacks. In order to ensure that each patient has full control over his identity information and personal health information, we leverage CP-ABE proposed in [1] and ABS proposed in [10] as the encryption primitive and the signature primitive in our possible solution. Our scheme realizes the same three levels of security and privacy requirement as the PSMPA scheme. All the members are also classified into three categories: the directly authorized physicians in the local hospital, the indirectly authorized physicians in the remote hospital and the unauthorized persons. We generally describe the possible solution and discuss its security in the following.

Before encrypting the PHRs, the patients divide the PHRs into patient's identity information m_1 and personal health information m_2. To achieve the goal of access control, the CP-ABE scheme in [1] is brought in. The patients can use two different access tree T_1 and T_2 to encrypt m_1 and m_2 into CT_1 and CT_2 respectively. The set of leaf nodes in T_2 does not contain the attribute of the hospital where the physician works, while the access tree T_1 contains. The patient can define the root node of T_1 as an "AND" gate with two children: one is T_2 and the other is a leaf that is associated with the attribute of the hospital where the physician works. For example, if patient P is registered in hospital A, he can specify the attribute "hospital=A" as the leaf node of the root. In this way, only the directly authorized physicians working in hospital A whose attributes satisfy T_2 are able to decrypt the ciphertexts (CT_1, CT_2) and get the plaintext (m_1, m_2) simultaneously, while the indirectly authorized physicians working in other hospital whose attributes satisfy T_2 only can decrypt CT_2 and get m_2. The unauthorized persons whose attributes can not satisfy T_2 will obtain nothing. Through constructing the two different access tree, we realize the fine-grained access control to patient's identity information and personal health information.

As we all know, encryption offers confidentiality and signature provides authenticity, one can perform encryption and signing sequentially to achieve this. Once receiving the PHRs uploaded by someone, the storage server in hospital must check their authenticity. Traditional digital signature can undertake this task, but the patient's identity will be exposed to the ones who are not desired by the patient. In [10], Rao *et al.* constructed a key-policy ABS scheme with constant-size signature to achieve signer privacy. A valid ABS attests to the fact that "a single user, whose attributes satisfy the predicate, endorsed the message" and provides the public verifiability. The public just knows the signature comes from people who satisfy certain criteria like that they should possess some specific attributes. In our possible solution, the patients leverage ABS to sign the ciphertexts $CT_1||CT_2$ before generated from CP-ABE and output the corresponding signature σ. Finally, the patients produce the tuple (CT_1, CT_2, σ) and upload it to the storage server in the local hospital. Receiving the tuple, the storage server executes the verify algorithm of ABS. If the signature passes the validation, the server stores the tuple. Otherwise, the server rejects it. Because of the utilization of ABS, the new PHRs sharing scheme achieves the function of anonymous authentication successfully.

In the new scheme, not all the physicians working in the same hospital as the patient P can recognize P's actual identity, except the ones whose attributes satisfy T_2. Therefore, the collusion attack described in Section 3 does not exist in our scheme. Furthermore, since the PHRs received by indirectly authorized physicians derive from the patients directly instead of the directly authorized physicians, our scheme also does not suffer from the forgery attack as PSMPA.

In this section, we provide a possible solution to avoid the above two attacks. To ensure that each patient has full control over his identity information and personal health information, we leverage CP-ABE proposed in [1] and ABS proposed in [10] as the encryption primitive and the signature primitive in our possible solution. Our scheme realizes the same three levels of security and privacy requirement as the PSMPA scheme. As shown in Figure 1, All the members are also classified into three categories: the directly authorized physicians such as Bob in the local healthcare provider, the indirectly authorized physicians such as Jack, Tom and Jim in the remote healthcare providers and the unauthorized persons such as Black. We generally describe the possible solution which is consisted of five phases and discuss its security in the following.

Setup. The algorithm takes 1^l as input, where l is the security parameter. It outputs public parameters and y as the master key for the central attribute authority. This algorithm is the same as the setup algorithm in the PSMPA scheme.

Key Extract. As the ABS and CP-ABE involved, both patients and physicians request their own attribute keys for an attribute set in this algorithm. If someone is qualified to be issued with sk_D for some attributes, the attribute authority produces sk_D for him.

Encrypt-Sign. Before encrypting the PHRs, the patients firstly divide the PHRs into patient's identity information m_1 and personal health information m_2. Secondly, they choose two different access tree T_1 and

T_2 as the corresponding access policy of the plaintexts m_1 and m_2. The set of leaf nodes in T_2 does not contain the attribute of the hospital where the physician works, while the access tree T_1 contains. The patient can define the root node of T_1 as an "AND" gate with two children: one is T_2 and the other is a leaf that is associated with the attribute of the hospital where the physician works. For example, if patient P is registered in hospital A, he can specify the attribute "hospital=A" as the leaf node of the root. Taking the two access tree and public parameters as input, the encryption algorithm encrypt m_1 and m_2 into CT_1 and CT_2 respectively.

Finally, to provide authenticity, the patients need to claim that they possess some specific attributes which the healthcare provider requires. Taking the corresponding attribute keys and public parameters as input, the signing algorithm signs the ciphertexts $CT_1||CT_2$ and outputs the signature σ. In this way, a tuple (CT_1, CT_2, σ) can be constructed and uploaded to the storage server in the local hospital.

Verify. Once receiving the PHRs uploaded by someone, the storage server in hospital executes the verify algorithm of ABS and decides whether the signer possesses the attributes as they claimed in the signature σ. If the signature σ passes the validation, the server stores the tuple. Otherwise, the server rejects it.

Decrypt. When the physicians issue a request to the server, it returns the corresponding tuple. Receiving the tuple, the directly authorized physicians working in local healthcare provider whose attributes satisfy T_2 decrypt the ciphertexts (CT_1, CT_2) and get the plaintexts (m_1, m_2) simultaneously through executing the decrypt algorithm of CP-ABE, while the indirectly authorized physicians working in remote healthcare provider whose attributes satisfy T_2 only can decrypt CT_2 to get m_2 using their attribute keys. The unauthorized persons whose attributes can not satisfy T_2 will obtain nothing.

In our scheme, we treat the CP-ABE proposed in [1] as the encryption primitive. For purpose of realizing collusion-resistance, Bethencourt *et al.* [1] embeds independently chosen secret shares into the ciphertext such that the attacks can not combine their attribute keys to satisfy the access tree. Thus, not all the physicians working in the local healthcare provider can recognize the patient's actual identity, except the ones whose attributes satisfy T_2. However, in the PSMPA scheme, the fact that all the directly authorized physicians working in the local healthcare provider can decrypt the patient's identity causes the collusion attack. Therefore, our new scheme can resist the collusion attack between the directly authorized physicians and the indirectly authorized physicians.

Furthermore, since the PHRs received by indirectly authorized physicians derive from the patients directly instead of the directly authorized physicians and we do not hide the patient's actual identity by randomizing it, our scheme does not suffer from the forgery attack as PSMPA. In summary, the PHRs sharing scheme proposed above can be regarded as a possible solution for the PSMPA scheme.

5 Conclusions

In this paper, we discuss two important flaws in the patient self-controllable multi-level privacy-preserving cooperative authentication scheme. Exploiting collusion attack and forgery attack, we specify that the scheme doesn't possess the feature of identity privacy as they have claimed and there exists a flawed design during the transcript simulation. In the end, we establish an improved PHRs sharing scheme as a remedy solution through incorporating CP-ABE and ABS. A concrete description of the proposed scheme will be given in the future work.

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China under Grant 61003230, Grant 61370026, and Grant 61202445, in part by the Fundamental Research Funds for the Central Universities under Grant ZYGX2013J073, and in part by the Applied Basic Research Program of Sichuan Province under Grant 2014JY0041.

References

[1] J. Bethencourt, A. Sahai, and B. Waters, "Ciphertext-policy attribute-based encryption," in *IEEE Symposium on Security and Privacy (SP'07)*, pp. 321–334, Berkeley, USA, 2007.

[2] M. Chase and S. S. Chow, "Improving privacy and security in multi-authority attribute-based encryption," in *Proceedings of the 16th ACM Conference on Computer and Communications Security (CCS'09)*, pp. 121–130, Chicago, USA, 2009.

[3] V. Goyal, O. Pandey, A. Sahai, and B. Waters, "Attribute-based encryption for fine-grained access control of encrypted data," in *Proceedings of the 13th ACM Conference on Computer and Communications Security (CCS'06)*, pp. 89–98, Alexandria, USA, 2006.

[4] L. Guo, C. Zhang, J. Sun, and Y. Fang, "Paas: A privacy-preserving attribute-based authentication system for ehealth networks," in *IEEE 32nd International Conference on Distributed Computing Systems (ICDCS'12)*, pp. 224–233, Macau, China, 2012.

[5] C. C. Lee, P. S. Chung, and M. S. Hwang, "A survey on attribute-based encryption schemes of access control in cloud environments.," *International Journal of Network Security*, vol. 15, no. 4, pp. 231–240, 2013.

[6] M. Li, S. Yu, K. Ren, and W. Lou, "Securing personal health records in cloud computing: Patient-centric and fine-grained data access control in multi-owner settings," in *6th International ICST Conference on Security and Privacy in Communication Networks*, pp. 89–106, Singapore, 2010.

[7] M. Li, S. Yu, Y. Zheng, K. Ren, and W. Lou, "Scalable and secure sharing of personal health records in cloud computing using attribute-based encryption," *IEEE Transactions on Parallel and Distributed Systems*, vol. 24, no. 1, pp. 131–143, 2013.

[8] Q. Li, H. Xiong, F. Zhang, and S. Zeng, "An expressive decentralizing kp-abe scheme with constant-size ciphertext.," *International Journal of Network Security*, vol. 15, no. 3, pp. 161–170, 2013.

[9] X. Liu, J. Ma, J. Xiong, and G. Liu, "Ciphertext-policy hierarchical attribute-based encryption for fine-grained access control of encryption data.," *International Journal of Network Security*, vol. 16, no. 4, pp. 351–357, 2014.

[10] Y. S. Rao and R. Dutta, "Efficient attribute-based signature and signcryption realizing expressive access structures," *International Journal of Information Security*, pp. 1–29, 2015.

[11] S. Yu, C. Wang, K. Ren, and W. Lou, "Achieving secure, scalable, and fine-grained data access control in cloud computing," in *The 29th Conference on Computer Communications (INFOCOM'10)*, pp. 1–9, San Diego, USA, 2010.

[12] S. Yu, C. Wang, K. Ren, and W. Lou, "Attribute based data sharing with attribute revocation," in *Proceedings of the 5th ACM Symposium on Information, Computer and Communications Security (ASIACCS'10)*, pp. 261–270, Beijing, China, 2010.

[13] J. Zhou, X. Lin, X. Dong, and Z. Cao, "Psmpa: Patient self-controllable and multi-level privacy-preserving cooperative authentication in distributed m-healthcare cloud computing system," *IEEE Transactions on Parallel and Distributed Systems*, vol. 26, no. 6, pp. 1693–1703, 2014.

Yang Zhao is a Ph.D. Candidate at the School of Computer Science and Engineering, University of Electronic Science and Technology of China. His research interests are in the area of networking security and e-commerce protocol.

Feng Yue received his B.S. degree in the School of International Education, Henan University of Science and Technology (HAUST) in 2008. He is currently pursing his M.S. degree in the School of Computer Science and Engineering, (UESTC). His research interests include: cryptography and information security.

Songyang Wu is an associate professor at The Third Research Institute of Ministry of Public Security, China. Vice director. He received his Ph.D. Degree in computer Science from TongJi University, China in 2011. His current research interests are in information security, cloud computing and digital forensics.

Hu Xiong is an associate professor at University of Electronic Science and Technology of China (UESTC). He received his Ph.D degree from UESTC in 2009. His research interests include: cryptography and network security.

Zhiguang Qin is the dean and professor in the School of Computer Science and Engineering, University of Electronic Science and Technology of China (UESTC). He received his PH.D. degree from UESTC in 1996. His research interests include: information security and computer network.

An Improved Anonymous Password Authentication Scheme Using Nonce and Bilinear Pairings

Jie Ling, Guangqiang Zhao
(Corresponding author: Guangqiang Zhao)
Faculty of Computer, Guangdong University of Technology, Guangzhou 510006, China
(Email: gqzh88@126.com)
(Received Jan. 09, 2015; revised and accepted Apr. 27 & June 4, 2015)

Abstract

In 2013, Li et al. pointed out the security problems of Chen's password authentication scheme. they proposed an enhanced smart card based remote user password authentication scheme and claimed their scheme is secure against replay attacks, forgery attacks. In this paper, we state that the scheme is vulnerable to user impersonation attack. It also suffers from user anonymity violation and clock synchronization problem. Furthermore, an improved anonymity enhancement password authentication scheme using nonce and bilinear pairing is proposed. The analysis shows that the proposed scheme is more suitable for applications with high security requirements.
Keywords: Anonymity, authentication, bilinear pairing, clock synchronization nonce

1 Introduction

With the rapid development of network technologies, the client/server based service architecture has become the major service mode for Internet. It enables a single computer to serve a huge amount of clients which are dispersed over different regions around the world [6]. More and more services such as online banking, online trading and online money transfers etc. are provided by the internet. However, almost all of them are operated through the open networks, which may be intrusion by a malicious adversary or illegal users and lead to the private information leakage and properties missing of legal users [4, 20]. Hence, a considerable amount of researches have been carried out to enhance the security of communications over insecure networks. Password authentication scheme using smart card becomes one of the most widely used methods. Although quite a number of remote user authentication schemes with smart cards have been proposed, none of them can solve all possible problems and withstand all possible attacks [8]. Zhu [21] presented an authentication scheme for wireless environments which was proved to be insecure by Lee in 2006, and Lee proposed a new enhanced one [10].

In 2008, Liao put forward a dynamic ID based remote user authentication scheme which could not withstand impersonation attacks and reflection attacks [14]. It was insecure when a user could log in the remote server successfully with a random password, Xu [19] proposed a password authentication scheme based on smart card in 2009 and claimed it is secure. However, Sood [17] and Song [15] proved that the scheme was vulnerable to impersonation and internal attacks and proposed their improved schemes respectively. Nevertheless, Chen et al. [3] found that there still exist security problems, where mutual authentication is not achieved in the scheme of Sood and offline guessing attacks cannot be resisted in the scheme of Song. Then they proposed an improved password authentication and key agreement scheme. Unfortunately, Saru et al. [9] pointed out that Chen's scheme fails to resist impersonation attack and insider attack, it does not provide important features such as user anonymity and confidentiality to air messages. Later, Li et al. [12] also showed that Chen et al.'s scheme cannot ensure forward secrecy and the password change phase of the scheme is inefficient when the users update their passwords, in order to eliminate these problems, they proposed a modified smart card based user authentication scheme and claimed it is more secure. However, we find that Li et al.'s scheme is vulnerable to user impersonation attack, insider attack. Besides, it also suffers from user anonymity violation and clock synchronization problem. Furthermore, we propose an anonymous password authentication scheme based on smart card using nonce and bilinear pairings. We prove it can overcome the above security flaws and is more suitable for practical applications.

The rest of the paper is organized as follows: in Section 2, we introduce the notions used in this paper and bilinear pairings knowledge which is the security of our enhanced scheme. In Section 3, we provide a brief review of Li's scheme and demonstrate the security weakness of

the scheme. Meanwhile, our proposed scheme and corresponding scheme analysis are presented in Section 4, respectively. At last, we draw our conclusions in Section 5.

2 Preliminaries

2.1 Notations

The notations used through out this paper are summarized as follows:

- U_i: the ith user.
- SC: the smart card.
- S: the authentication server.
- ID_i: the identity of U_i.
- PW_i: the password of U_i.
- x: the master secret key hold by server S.
- $\triangle T$: the maximum transmission delay.
- p,q: two large prime numbers that satisfy $p = 2q + 1$.
- Z_q: the ring of integers modulo q.

2.2 Bilinear Pairings

Suppose G_1 is an additive cyclic group generated by P, Whose order is a prime q,and G_2 is a multiplicative cyclic group of the same order.A map e: $G_1 \times G_1 \Rightarrow G_2$ is called a bilinear mapping if it satisfies the following properties:

1) Bilinear:$e(aP,bQ) = e(P, Q)^{ab}$ for all $P,Q \in G_1$ and $a,b \in Z_q$.
2) Non-degenerate: there exist $P,Q \in G_1$ such that $e(P, Q) \neq 1$.
3) Computable: there is an efficient algorithm to compute $e(aP,bQ)$ for all $P,Q \in G_1$.

We note that G_1 is the group of points on an elliptic curve and G_2 is a multiplicative subgroup of a finite field. Typically, the mapping e will be derived from either the Weil or the Tate pairing on an elliptic curve over a finite field.

3 Review and Discussion

Li's scheme consists of Registration phase, Login phase, Authentication phase and Password change phase. The detailed steps of these phases are shown as follows and also in Figure 1.

3.1 Registration Phase

Step 1. U_i chooses his identity ID_i and password PW_i and submits them to S via a secure channel.

Step 2. S computes $A_i \doteq h(ID_i||PW_i)^{PWi} \bmod p$. $B_i = h(ID_i)^{x+PW_i} \bmod p$.

Step 3. S stores $\{A_i,B_i,h(),p,q\}$ in a SC and issues the SC to U_i via a secure channel.

3.2 Login Phase

Step 1. U_i inserts SC into a card reader and inputs his identity ID_i and password PW_i.

Step 2. SC computes ${A_i}^* \doteq h(ID_i||PW_i)^{PWi} \bmod p$, and compares ${A_i}^*$ with A_i, where A_i is stored in SC. If they are not equal, it means the user entered a wrong password and SC terminates the session. If $A_i = {A_i}^*$, SC performs the following steps.

Step 3. SC chooses a random number $\alpha \in {Z_q}^*$ and computes: $C_i = B_i \ / \ h(ID_i)^{PWi} \bmod p$, $D_i = h(ID_i)^{\alpha} \bmod p$,$M_i = h(ID_i \| C_i \| D_i \| T_i)$, where T_i is the current time.

Step 4. SC sends the login request message $\{ID_i,D_i,M_i,T_i\}$ to S.

3.3 Authentication Phase

Step 1. S checks that the ID_i is valid and that ${T_i}^*$ - $T_i \leq \triangle T$, where ${T_i}^*$ is the time the login request was received. If either or both are invalid, the login request is rejected.

Step 2. S computes ${C_i}^* = h(ID_i)^x \bmod p$, $M_i^* = h(ID_i \ \| \ C_i^* \ \| \ D_i \ \| \ T_i)$.

Step 3. S compares ${M_i}^*$ with received M_i. If equal, the login request is accepted and U_i is authenticated by server S; otherwise, the login request is rejected.

Step 4. S generates a random number $\beta \in {Z_q}^*$ and computes:$V_i = h(ID_i)^{\beta} \bmod p$, and the shared session key $sk = {D_i}^{\beta} \bmod p$.

Step 5. S gets the current time stamp T_S, and computes $M_S = h(ID_i \| {C_i}^* \| V_i \| sk \| T_S)$, and sends the mutual-authentication message $\{ID_i,V_i,M_S,T_S\}$ to U_i.

Step 6. Upon receiving the message, U_i checks ID_i and compares T_S with ${T_S}^*$, where ${T_S}^*$ is the time the mutual authentication message was received. If ID_i is valid and ${T_S}^*$ - $T_S \leq \triangle T$, U_i performs the following steps.

Step 7. U_i computes: $sk^* = V_i^{\alpha} \bmod p$, $M_S^* = h(ID_i \ \| \ C_i \ \| \ V_i \ \| \ sk^* \ \| \ T_S)$, and compares ${M_S}^*$ with the received M_S. If they are not equal, the session is terminated. On the contrary, if ${M_S}^* = M_S$, the server S is authenticated by the user U_i.

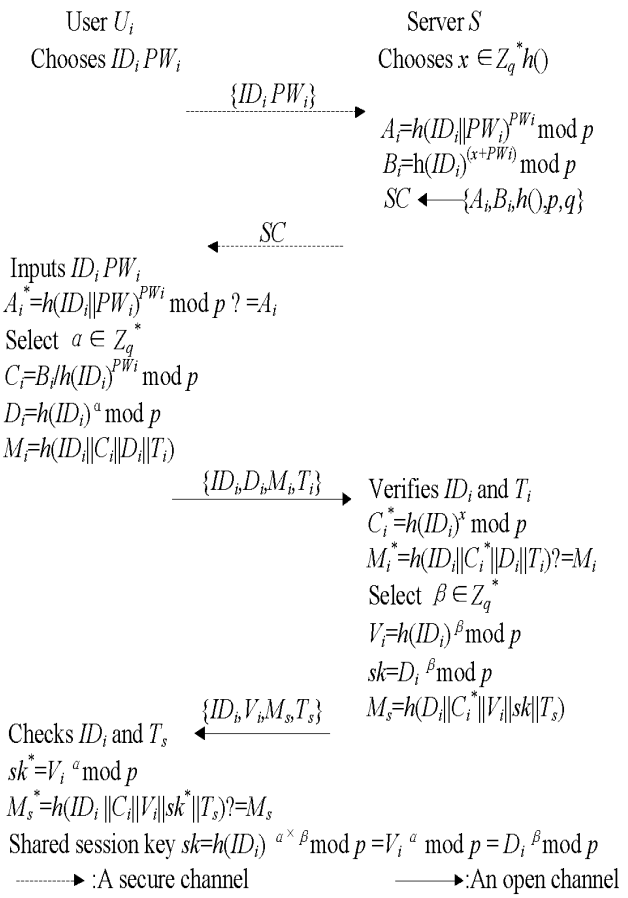

Figure 1: Li's scheme

At last, the user U_i and the server S share an agreed session key $sk = {D_i}^{\alpha\beta} \bmod p$.

3.4 Password Change Phase

This phase is invoked whenever U_i wants to change his passwordPW_i with a new password ${PW_i}^{new}$, and it can be finished without communicating with the server S.

Step 1. U_i inserts his/her smart card into a card reader and submits his/her identity ID_i, password PW_i, and requests to change the password.

Step 2. SC computes ${A_i}^* = h(ID_i||PW_i)^{PWi} \bmod p$, and compares ${A_i}^*$ with A_i, where A_i is stored in SC. If they are not equal, SC rejects the password change request. On the contrary, if ${A_i}^* = A_i$, the user is asked to key a new password ${PW_i}^{new}$.

Step 3. SC computes ${A_i}^{new} = h(ID_i||{PW_i}^{new})^{PW_i new} \bmod p, {B_i}^{new} = B_i \times h(ID_i)^{PW_i new}/h(ID_i)^{PW_i} \bmod p$.

Step 4. SC replaces A_i,B_i with ${A_i}^{new}$, ${B_i}^{new}$, respectively.

3.5 Cryptanalysis of Li et al. Scheme

3.5.1 User Impersonation Attack

During login phase, U_i sends login message $\{ID_i, D_i, M_i, T_i\}$ to S, An attacker U_a can easily obtain the ID_i of U_i by intercepting any login request between U_i and S. Then in near future, U_a can impersonate U_i to cheat S as follows:

1) U_a sends the registration request message ID_i, PW_a, where ID_i is the identity of U_i and PW_a is chosen by U_a as his password.

2) S sends the SC which contains $\{A_a,B_a,h(),p,q\}$ to U_a,where $A_a = h(ID_i||PW_a)^{PWa} \bmod p$,$B_a = h(ID_i)^{x+PWa} \bmod p$.

3) U_a extracts values $\{A_a,B_a,h(),p,q\}$from his/her smart card and computes C_i=B_a $/h(ID_i)^{PWa} \bmod p = h(ID_i)^x \bmod p$.

4) U_a chooses a random number $a^* \in Z_q^*$ and computes: $D_a = h(ID_i)^{a*}$, M_a= $h(ID_i||C_i||D_a||T_a)$, where T_a is the current time of U_a.

5) U_a sends the login request $\{ID_i,D_a,M_a,T_a\}$ to S.

It is easy to see that, S will of course accept it as a legal user because of the reasons:

1) It contains valid identity ID_i of U and the fresh timestamp T_a.

2) The equivalence ${M_a}^* = M_a$ holds since ${M_a}^* = h(ID_i||{C_i}^*||D_a||T_a)$ where ${C_i}^*= C_i = h(ID_i)^x \bmod p$.

S accept the adversary U_a and sends the response $\{ID_i,V_i,M_S,T_S\}$,upon the adversary U_a receiving the response message, just ignore it and computes the session key $sk = {V_i}^{a*}$.

3.5.2 Server Impersonation Attack

Here we move one step forward from the above user impersonation attack. Assume that the attacker possessing ID_i and $C_i = h(ID_i)^x \bmod p$ corresponding to U can impersonate S to cheat U_i as explained below:

1) Suppose U_i sends the login request $\{ID_i,D_i,M_i,T_i\}$ to S.

2) The attack intercepts and blocks $\{ID_i,D_i,M_i,T_i\}$ from reaching up to S, The attacker generates a random number $\beta \in {Z_q}^*$, and computes $V_i = h(ID_i)^\beta \bmod p$,$sk = {D_i}^\beta \bmod p$. S gets the current time stamp T_S, and computes $M_S = h(ID_i||{C_i}^*||V_i||sk||T_S)$, and sends the mutual authentication message $\{ID_i,D_i,M_S,T_S\}$ to U_i.

The message will pass the verification test at U_i because follows:

1) It contains the valid identity ID_i of U_i and fresh timestamp T_S.

2) The equivalence $M_S{}^* = M_S$ holds due to the fact that $sk^* = (V_i)^{\alpha} \bmod p = (D_i)^{\beta} \bmod p = D_i{}^{\alpha\beta} \bmod p$, $M_S{}^* = h(ID_i \| C_i \| V_i \| sk^* \| T_S) = M_S$.

3.5.3 Inside Attack

Password authentication is the most important and convenient protocol for verifying users to get the system's resources.If the password of a user can be derived by the server in the registration protocol, it is called the insider attack; it is a common practice in the real world that many users use the same passwords to access different servers for their convenience without remembering different passwords for different servers. However, the security of Li's authentication scheme relies on the secrecy of his password. Moreover, disclosure of users passwords to anyone is risky. Li skip this important aspect while building the registration phase of their scheme. Users submit the registration request message $\{ID_i, PW_i\}$ consisting their plaintext passwords to S. Therefore, malicious privileged insiders at S have direct access to users passwords PW and they can misuse them to impersonate the legal users or craft other harms.

3.5.4 Clock Synchronization Problem

Remote user authentication schemes employing timestamps to provide message freshness may still suffer from replay attacks as the transmission delay is unpredictable in existing networks. In addition, clock synchronization is difficult and expensive in existing network environments, especially in wireless and mobile networks and distributed networks [5]. Hence, these schemes employing the timestamp mechanism to resist replay attacks are not suitable for mobile applications [2, 7]. In He's scheme, this principle is violated.

3.5.5 Failure of Preserving User Anonymity

Most of the password authentication protocols are based on static identity, which can be used by the attacker to trace and identify the different requests belonging to the same user. On the other hand, the dynamic identity based authentication protocols are more suitable to e-commerce applications [16, 13], for they provide multi-factor authentication based on the identity, password, and smart card. In many cases such as secret online-order placement electronic auditing and electronic voting etc. it is very important to preserve user privacy. In Li's scheme, the user identity ID_i is transmitted in plaintext, which may leak the identity of the logging user once the login messages were eavesdropped. That is to say, without employing any effort an adversary can distinguish and recognize the particular transactions performed by the specific user U. Moreover, the user identity ID_i is static in all the login phases, which may facilitate the attacker to trace out the different login request messages belonging to the same user and to derive some information related to the user U_i. In summary, neither initiator anonymity nor initiator un-traceability can be preserved in their scheme [18].

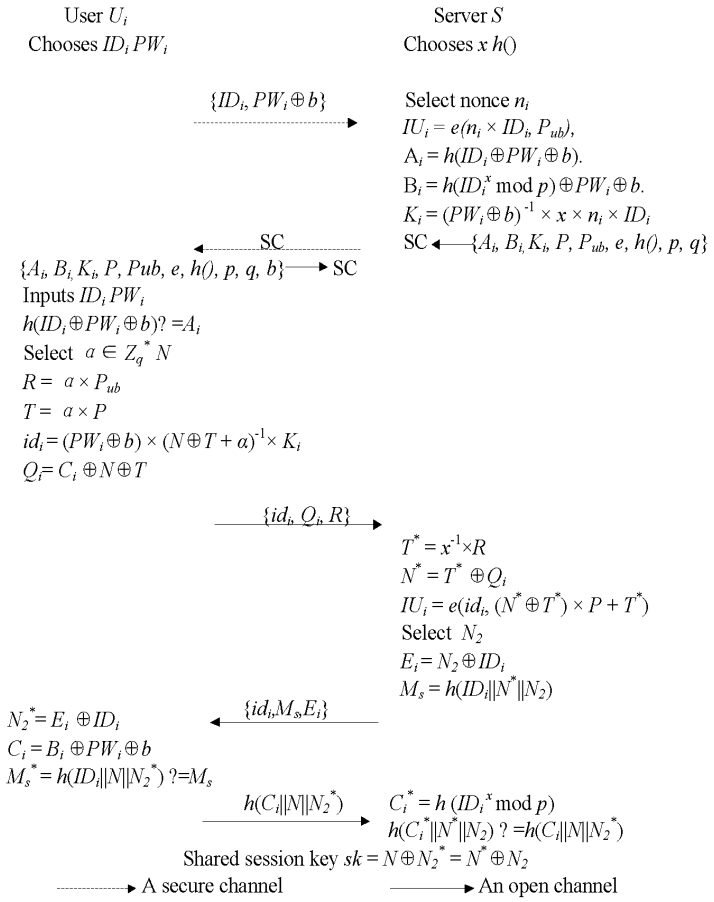

Figure 2: The proposed scheme

4 Our Proposed Scheme

In this section, we use bilinear pairings and nonce to propose an enhancement on Li's scheme that can withstand the security flaws described in previous sections. The proposed scheme performs as follows, and it is also shown in Figure 2.

4.1 The Setup Phase

Let G_1 be an additive cyclic group of a prime order q, and G_2 be a multiplicative cyclic group of the same order. Let P be a generator of G_1,e: $G_1 \times G_1 \Rightarrow G_2$ be a bilinear mapping and h:$\{0,1\} \Rightarrow G_1$ be a cryptographic one-way hash function which maps a string to a point of the additive cyclic group G_1, The server choose a secret key x and computes the corresponding public key $P_{ub} = x \times P$. The server publishes the system parameters $\{ G_1, G_2, e, q, P, P_u b, h()\}$and keeps x secret.

4.2 The Registration Phase

Step 1. U_i chooses ID_i, PW_i, and a random number b, then computes $PW_i \oplus b$ and submits $\{ID_i, PW_i \oplus b\}$ to S via a secure channel.

Step 2. Upon receiving the register message $\{ID_i, PW_i \oplus b\}$, S checks the uniqueness of ID_i in Table 1, if ID_i is in Table 1, it means the identity has been registered before. then U_i will be informed an illegal ID_i and asked to choose a new one, if not, S chooses a random nonce n_i, computes $K_i = 1/(PW_i \oplus b) \times x \times n_i \times ID_i$, $IU_i = e(n_i \times ID_i, P_{ub})$, $A_i = h(ID_i \oplus PW_i \oplus b)$, $B_i = h(ID_i^x \bmod p) \oplus PW_i \oplus b$.

Step 3. S stores $\{A_i, B_i, K_i, P, P_{ub}, e, h(), p, q\}$ in a SC and issues the SC to U_i via a secure channel and S stores $\{IU_i, ID_i\}$ in Table 1 which in a secure database.

Step 4. U_i inserts b into SC, that is, SC contains $\{A_i, B_i, K_i, P, P_u b, e, h(), p, q, b\}$.

Table 1: Index of U and its related identity

Index of the users identity	User identity
IU_1	ID_1
IU_2	ID_2
IU_3	ID_3
$\cdots$	$\cdots$

4.3 The Login Phase

Step 1. U_i inserts his smart card into a card reader and inputs ID_i and PW_i.

Step 2. SC computes and compares $h(ID_i \oplus PW_i \oplus b)$ with A_i. If not equal, it means enter a wrong password or an illegal identity, the smart card terminates the session. If $h(ID_i \oplus PW_i \oplus b) = A_i$, SC performs the following steps.

Step 3. SC chooses a random number $\alpha \in {Z_q}^*$ and computes $R = \alpha \times P_{ub}$, $T = \alpha \times P$, besides, chooses a nonce N, computes the temporary identity of U_i, $id_i = PW_i \oplus b \times 1/(N \oplus T + \alpha) \times K_i$, $Q_i = N \oplus T$, then SC sends the message $\{id_i, Q_i, R\}$ to the server.

4.4 The Authentication Phase

Step 1. Upon receiving the message, S computes $T^* = 1/x \times R$, $N^* = Q_i \oplus T^*$, $IU_i = e(id_i, (N^* \oplus T^*) \times P + T^*)$,Then S search for ID_i related to IU_i in Table 1, if fails, S terminated the session, otherwise, S gets ID_i and performs steps below:

Step 2. S computes $E_i = ID_i \oplus N_2$, where N_2 is a random nonce in sequence. $M_S = h(ID_i \| N^* \| N_2)$ and sends the message $\{id_i, M_S, E_i\}$ to U_i.

Step 3. After received the message, U_i checks id_i and computes ${N_2}^* = ID_i \oplus E_i$, ${M_S}^* = h(ID_i \| N \| {N_2}^*)$ and compares ${M_S}^*$ with M_S, if they are equal, S is authenticated by U_i. U_i computes $C_i = B_i \oplus PW_i \oplus b$ and sends S the message $h(C_i \| N \| {N_2}^*)$.

Step 4. S computes ${C_i}^* = h({ID_i}^x \bmod p)$ and verifies $h({C_i}^* \| N^* \| N) = h(C_i \| N \| {N_2}^*)$. If equal, S believes U_i is authenticated.

Step 5. SC and S compute the shared session key $sk = N \oplus {N_2}^* = N^* \oplus N_2$.

4.5 The Password Change Phase

Step 1. U_i inserts SC into a terminal and submits ID_i, PW_i, SC computes and compares $h(ID_i \oplus PW_i \oplus b)$ with A_i, if equal, the users is asked for a new password ${PW_i}^{new}$.

Step 2. SC computes ${A_i}^{new} = h(ID_i \oplus {PW_i}^{new} \oplus b)$, ${K_i}^{new} = K_i \times (PW_i \oplus b) \times 1/({PW_i}^{new} \oplus b)$, ${B_i}^{new} = B_i \oplus PW_i \oplus {PW_i}^{new} \bmod p$.

Step 3. SC replaces A_i, K_i, B_i with ${A_i}^{new}$, ${K_i}^{new}$, ${B_i}^{new}$ respectively.

4.6 Correctness, Security and Performance

4.6.1 Correctness

If S received the message $\{id_i, Q_i, R\}$, S computes the index of the identity of U_i based the equation $IU_i = e(id_i, (N^* \oplus T^*) \times P + T^*)$ of Step1 of the authentication phase holds, which is verified as below:

$$
\begin{aligned}
& e(id_i, (N^* \oplus T^*) \times P + T^*) \\
= \; & e((PW_i \oplus b) \times 1/(N \oplus T + \alpha) \times K_i, \\
& \quad (N^* \oplus T^*) \times P + T^*) \\
= \; & e((PW_i \oplus b) \times 1/(N \oplus T + \alpha) \times K_i, \\
& \quad (N^* \oplus T^*) \times P + 1/x \times \alpha \times x \times P) \\
= \; & e((PW_i \oplus b) \times 1/(N \oplus T + \alpha) \times K_i, \\
& \quad ((N^* \oplus T^*) + \alpha) \times P) \\
= \; & e((PW_i \oplus b) \times 1/(N \oplus (\alpha \times P) + \alpha) \times K_i \times N^* \\
& \quad \oplus (1/x \times \alpha \times x \times P) + \alpha), P) \\
= \; & e((PW_i \oplus b) \times 1/(PW_i \oplus b) \times x \times n_i \times ID_i, P) \\
= \; & e(x \times n_i \times ID_i, P) \\
= \; & e(n_i \times ID_i, x \times P) \\
= \; & e(n_i \times ID_i, P_{ub}) \\
= \; & U_i.
\end{aligned}
$$

4.6.2 Security

We analyze the security of our enhanced scheme and compare it with other related schemes. The functionality comparison of our proposed scheme and other related works

Table 2: Functionality comparisons

schemes	S1	S2	S3	S4	S5	S6	S7	S8
Xu et al. [19]	Y	Y	N	N	Y	N	N	N
Sood et al. [17]	N	N	N	N	Y	N	N	N
Song [15]	Y	Y	N	N	Y	N	N	N
Chen et al. [3]	N	N	N	N	Y	N	N	N
Li et a [12]	N	N	N	N	Y	Y	Y	N
Ours	Y	Y	Y	Y	Y	Y	Y	Y

is summarized in Table 2, from which we can see that the proposed scheme is more secure than other related schemes. We demonstrate this as below:

S1: Preventing User Impersonation Attack.
This attack means that an adversary may try to intercepted the login messages $\{id_i,Q_i,R\}$ or forge a message to masquerade a legal user to cheat S. Unfortunately, it is impossible for the adversary to compute valid value $h(C_i\|N\|{N_2}^*)$ of Step3 in the authentication phase. Because the plaintexts of C_i,N and ${N_2}^*$ are not transmitted on the channel. Moreover, the adversary cannot compute C_i and ${N_2}^*$ based on $\{id_i,M_i,,E_i\}$ without knowing the secret key x of S and ID_i of U_i. Hence, our scheme can resist user impersonation attack.

S2: Preventing Server Spoofing Attack.
The adversary may attempt to cheat the requesting user U_i.However, it has to forge a valid response message $\{id_i,M_S,E_i\}$ after receiving message $\{id_i,Q_i,R\}$,due to $E_i = ID_i \oplus N_2$, and ID_i can only get through IU_i,IU_i=$e(id_i,(N^* \oplus T^*)\times P + T^*)$and T^*=$1/x \times R$, the adversary cannot computer IU_i without knowing the secret key x of S. Therefore, our proposed scheme can resist server spoofing attack.

S3: Preventing the Insider Attack.
The insider attack occurs when the user password is obtained by the server in the registration phase. Therefore, the users must conceal their passwords from the server to prevent this kind attack. In our enhanced scheme, the user sends the register message $\{ID,PW_i \oplus b\}$ to S, S cannot know the PW of U since the entropy of b is very large. Hence, the malicious adversary in the server cannot carry out this attack.

S4: User Anonymity and Intractability.
User anonymity requires that only the server knows the identity of the user with whom he is interacting, while any third party is unable to do this. User intractability requires that any adversary should be prevented from linking one unknown user interacting with the server to another transcript, that is to say, the adversary is not capable of telling whether he has seen the same user twice [11]. Our proposed scheme use bilinear pairings to protect user true identity. A secure login message is used for protect the user identity form disclosure. In the login phase of the scheme, the user U_i submits the masked identity id_i=$(PW_i \oplus b)\times 1/(N \oplus T + \alpha)\times K_i$, The attacker cannot compute the true identity of U_i based on id_i and IU_i =$e(id_i,(N^* \oplus T^*)\times P + T^*)$, because he cannot computes T^* without knowing the secret key x of S. Meanwhile, the temporary identity of U_i changes every time. Therefore, the true identity of U_i is protected. From the above analysis, we can see that our proposed protocol can provide the user anonymity and intractability.

S5: Preventing Replay Attacks.
The replay attack is when an attacker tries to imitate a legal user to log in to the server by resending the messages transmitted between U_i and S. In our proposed scheme, U_i first chooses a nonce N, computes id_i and send it to S. The second nonce N_2 is chosen by S and embedded in sk and E_i. The attacker may replay the previously used login request message and mutual authentication message to cheat the server or the user, However, he cannot replay an old login message $\{id_i,Q_i,R\}$ in login phase because he cannot compute the valid $h(C_i\|N\|{N_2}^*)$ without knowing ID_i and x.

S6: Perfect Forward Secrecy.
Perfect forward secrecy is an important property for session key distribution; which means that if a long term secret is compromised, the session key of previous sessions still cannot be derived. In our proposed scheme, the session key sk=$N \oplus {N_2}^*$, where N and N_2 are random nonce chosen by U and S, respectively. Also, N and N_2 change each time. Even if the attacker get the previous session key, he cannot computes the next session key between U and S. because N and N_2 are use only once by U and S.

S7: Prompt Detection of the Wrong Password.
Our proposed scheme uses the smart card password detection mechanism in the login phase. When U_i enters ID_i and PW_i, SC computes and compares $h(ID_i \oplus {PW_i}^{new} \oplus b)$ with A_i. If equals, SC performs the remaining steps of the login phase. If not. It

Table 3: Running time of different operations

Notations	Descriptions
T_e	pairing-based exponentiation, $1T_e \approx 11.20$ ms
T_h	hash operations, $1T_h \approx 432$ ms
T_m	elliptic curve scalar point multiplication, $1T_m \approx 6.38$ ms
T_s	encryption operations, $1T_s \approx 2826$ ms
T_d	decryption operations, $1T_d \approx 4357$ ms
T_b	bilinear pairing operation $1T_b \approx 20.01$ ms

Table 4: Performance comparisons

time	Computing	time	Running	time(ms)
schemes	Client	Server	Client	Server
Xu et al. [19]	$2T_e+4T_h$	$2T_e+4T_h$	1750.4	1750.4
Song [15]	$1T_s+4T_h$	$1T_e+1T_d+4T_h$	4554	6096.2
Sood et al. [17]	$3T_e+2T_m+3T_h$	$2T_e+1T_m+3T_h$	1342.36	1324.78
Chen et al. [3]	$2T_e+2T_m+4T_h$	$1T_e+1T_m+4T_h$	1763.16	1745.58
Li et al. [12]	$4T_e+1T_m+4T_h$	$3T_e+3T_h$	1779.18	1329.6
Ours	$4T_m+3T_h$	$1T_e+2T_m+3T_h+1T_b$	1321.52	1339.97

means the user entered an incorrect password SC terminates the session. Therefore, the wrong password will be detected timely at the beginning of the login phase by SC. It will not waste unnecessary extra communication and computation of S.

S8: Prevention of Clock Synchronization Problem. The timestamp is used to prevent replay attack in remote password authentication schemes. Meanwhile, it brings the clock synchronization problem. In our scheme, we discard the timestamp to avoid this problem. The enhanced scheme uses nonce not only prevents the clock synchronization problem but also can resist replay attack efficiently.

4.6.3 Performance

We evaluate the performance of our enhanced scheme and make comparisons with other related schemes. Since the login phase and the authentication phase are two principal parts of each password authentication scheme and should be performed in each session. We only consider the computation costs of these two phases. Let T_m, T_h, T_s, T_d, T_e, T_b be the time of multiplication/division operation, hashing operation, symmetric key encryption operation, symmetric key decryption operation, exponentiation and bilinear pairing operation respectively. The article [1] addressing the implementation of elliptic curve cryptosystems and bilinear on elliptic and estimated the running time of different cryptographic operations in Table 3. We estimate the executing time hash operation and encryption/decryption operation using Microsoft Visual C++ 6.0 software and C language in the environment of Windows XP operating system. The test data is less than 1024 bits. It shows the average time of hash operation is roughly 432 ms. The average executing time of encryption/decryption operation is 2826ms/4357 ms respectively. Table 4 shows the performance comparisons of our scheme and other related schemes. However, the proposed scheme is more secure and practical.

5 Conclusion

In this paper, we firstly showed that Li's scheme cannot resist user impersonation attack, server spoofing attack and insider attack, besides; it also suffers from user anonymity violation and clock synchronization problem. Then we proposed an anonymous password authentication scheme based on smart card using nonce and bilinear pairings, the enhanced scheme overcomes security weaknesses of the previous one. Compared with Li'scheme and other related scheme. Our improved scheme is as efficient as other related schemes and overcomes their weaknesses, which makes it is more secure and suitable for the practical applications.

Acknowledgments

This work is supported by the Project of Guangzhou Science and Technology(No.2014J4100201), and the Project of Guangdong Province Science and Technology(No.2013B040401017,2014A010103029), and the Key Program of the Natural Science Foundation of Guangdong Province(No.S2012020011071).

References

[1] X. Cao, W. Kou, and X. Du, "A pairing-free identity-based authenticated key agreement protocol with minimal message exchanges," *Information Sciences*, vol. 180, no. 15, pp. 2895–2903, 2010.

[2] C. Chang and C. Lee, "A secure single sign-on mechanism for distributed computer networks," *IEEE Transactions on Industrial Electronics*, vol. 14, no. 6, pp. 629–637, 2012.

[3] B. L. Chen, W. C. Kuo, and L. C. Wu, "Robust smart-card based remote user password authentication scheme," *International Journal of Communication Systems*, vol. 27, no. 2, pp. 377–389, 2014.

[4] L. Y. Gong, J. X. Pan, and B. B. Liu, "A novel one-time password mutual authentication scheme on sharing renewed finite random sub-passwords," *Journal of Computer and System Sciences*, vol. 79, no. 2, pp. 122–130, 2013.

[5] J. Han and D. Jeong, "A practical implementation of transparent clock for distributed measurement and control systems," *IEEE Transactions on Instrumentation and Measurement*, vol. 32, no. 5, pp. 433–439, 2010.

[6] M. S. Hwang, S. K. Chong, and T. Yu Chen, "DoS-resistant ID-based password authentication scheme using smart cards," *Journal of Systems and Software*, vol. 83, no. 2, pp. 163–172, 2010.

[7] S. Islam and Biswas G, "A more efficient and secure id-based remote mutual authentication with key agreement scheme for mobile devices on elliptic curve cryptosystem," *Journal of Systems and Software*, vol. 31, no. 2, pp. 1892–1898, 2011.

[8] M. Kumar, "A new secure remote user authentication scheme with smart cards," *International Journal of Network Security*, vol. 11, no. 2, pp. 88–93, 2010.

[9] S. Kumari and M. K. Khan, "Cryptanalysis and improvement of a robust smart-card-based remote user password authentication scheme," *International Journal of Communications Systems*, vol. 27, no. 2, pp. 377–389, 2014.

[10] C. C. Lee, M. S. Hwang, and I-En Liao, "Security enhancement on a new authentication scheme with anonymity for wireless environments," *IEEE Transactions on Industrial Electronics*, vol. 53, no. 5, pp. 1683–1687, 2006.

[11] C. C. Lee, C. H. Liu, and M. S. Hwang, "Guessing attacks on strong-password authentication protocol," *International Journal of Network Security*, vol. 15, no. 1, pp. 64–67, 2013.

[12] X. Li and J. W. Niu, "An enhanced smart card based remote user password authentication scheme," *Journal of Network and Computer Applications*, vol. 36, no. 5, pp. 1365–1371, 2013.

[13] Y. Lu, X. Yang, and X. Wu, "A secure anonymous authentication scheme for wireless communications using smart cards," *International Journal of Network Security*, vol. 17, no. 3, pp. 237–245, 2015.

[14] Bin du CS M.M, "Cryptanalysis of Lia-Lee-Hwang's dynamic ID scheme," *International Journal of Network Security*, vol. 6, no. 2, pp. 211–213, 2008.

[15] R. Song, "Advanced smart card based password authentication protocol," *Computer Standards and Interfaces*, vol. 32, no. 5, pp. 321–325, 2010.

[16] S. K. Sood, "An improved and secure smart card based dynamic identity authentication protocol," *International Journal of Network Security*, vol. 14, no. 6, pp. 39–46, 2012.

[17] S. K. Sood, A. K. Sarje, and K. Singh, "An improvement of Xu et al.'s authentication scheme using smart cards," *Proceedings of the Third Annual ACM Bangalore Conference*, vol. 11, no. 4, pp. 5–7, 2010.

[18] D. Wang, C. G. Ma, and M. S. Hwang, "Cryptanalysis of a remote user authentication scheme for mobile client-server environment based on ECC," *Information Fusion*, vol. 14, no. 2, pp. 498–503, 2013.

[19] J. Xu and D. G. Zhu, "An improved smart card based password authentication scheme with provable security," *Computer Standards and Interfaces*, vol. 31, no. 4, pp. 723–728, 2009.

[20] M. Zarepoor, P. Shamsolmoali, and M. A. Alam, "Establishing safe cloud: Ensure data security and performance evaluation," *International Journal of Electronics and Information Engineering*, vol. 2, no. 1, pp. 88–99, 2014.

[21] J. Zhu and J. Ma, "A new authentication scheme with anonymity for wireless environments," *IEEE Transactions on Consumer Electronics*, vol. 13, no. 2, pp. 84–91, 2010.

Jie Ling received Ph.D degree in computation mathematics from Sun Yat-sen University (China) in June 1988. He is a professor in computer science in Guangdong University of Technology.His current research interest fields include information security and Intelligent video processing technology.

Guang-qiang Zhao received his M.S degree in computer science from North China University of Water Resources and Electric Power (China) in June 2012, He is currently a Master degree candidate in Guangdong University of Technology (China). His current research interest fields include information security and cloud computing.

On the Security of Three Public Auditing Schemes in Cloud Computing

Yang Ming[1] and Yumin Wang[2]
(Corresponding author: Yang Ming)
School of Information Engineering, Chang'an University[1]
Xi'an, Shaanxi 710064, China
(Email: yangming@chd.edu.cn)
State Key Laboratory of ISN, Xidian University[2]
Xi'an, Shaanxi 710071, China
(Received Apr. 13, 2015; revised and accepted May 20 & June 4, 2015)

Abstract

Cloud computing provides a scalability environment for growing amounts of data and processes that work on various applications and services by means of on-demand self-services. It is necessary for cloud service provider to offer an efficient audit service to check the integrity and availability of the stored data in cloud. In this paper, we study three auditing schemes for stored data including the public auditing scheme with user revocation, the proxy provable data possession and the identity-based remote data possession checking. All three mechanisms claimed that their schemes satisfied the security property of correctness. It is regretful that this comment shows that an active adversary can arbitrary alter the cloud data to generate the valid auditing response which can pass the verification. Then, we discussed the origin of the security flaw and proposed methods to remedy the weakness. Our work can help cryptographers and engineers design and implement more secure and efficient auditing mechanism in the cloud.

Keywords: Bilinear pairings, cloud computing, provable data possession, storage auditing

1 Introduction

In recent years, cloud computing is a promising computing model that enables convenient and on-demand network access to a shared pool of computing resources [12]. It provided a flexible, dynamic, resilient and cost effective infrastructure for both academic and business environments, it rapidly expands as an alternative to conventional office-based computing. Cloud computing offers various types of services, including, Infrastructure as a Service (IaaS, Amazon's Elastic Cloud), Platform as a Service (PaaS, Microsoft Azure) and Software as a Service (SaaS, Google Web Mail Service) [1]. The cloud storage is an important service of cloud computing, which allows data owners to move data from their local computing system to the cloud. Thus, it relieves the burden for storage management and maintenance for the data owners.

Although cloud storage service (CSS) provided many appealing benefits for the user, it also prompts a number of security issues, because the user data or archives are stored into an uncertain storage pool outside the enterprises. Firstly, data owners would worry their data could be mis-used or accessed by unauthorized users. Secondly, the data owners would worry their data could be lost in the cloud. Therefore, it is necessary for cloud service providers to offer an efficient audit service to check the integrity and availability of the stored data in the cloud.

The traditional cryptographic technologies for data integrity and availability, based on hash functions and signature schemes, cannot work on the outsourced data without a local copy of data. In addition, it is not a practical solution for data validation by downloading them due to the expensive communications, especially for large size files. Therefore, it is crucial to realize public auditability, so that data owners may resort to a third party auditor (TPA), who has expertise and capabilities that a common user does not have, for periodically auditing the outsourced data.

To achieve this goal, two novel approaches called provable data possession (PDP) [2] and proofs of retrievability (POR) [7]. The new techniques are such a probabilistic proof technique for a storage provider to prove the integrity and ownership of user's data without download data. In 2007, Ateniese et al. [2] firstly proposed the PDP model for ensuring possession of files on untrusted storages and provided an RSA-based scheme for the static case They also proposed a publicly verifiable version, which allows anyone, not just the owner, to challenge the servers for data possession. In 2008, Ateniese et al. [3] proposed a dynamic PDP called scalable PDP which can support dynamic data operations. The new

scheme is constructed using cryptographic Hash function and symmetric key encryption, but the number of updates and challenges is limited and need to be prefixed and block insertion is not allowed. Since then, Erway et al. [6] introduced two dynamic PDP schemes based on a hash function tree. In 2007, Juels [7] presented a POR method to audit the integrity of data. However, it relies largely on preprocessing steps the user conducts before sending the data to cloud service provider (CSP), which prevent any efficient extension to update data. Shacham and Waters [11] proposed the compact POR (CPOR) scheme built from BLS signature [4] with proofs of security. They also use publicly verifiable homomorphic linear authentications that are built form provable secure BLS signature. Wang et al. [22] presented a dynamic scheme with by integrating above CPOR scheme and Merkle Hash Tree in DPDP. Wang et al. [17] provided a privacy-preserving auditing protocol. This scheme achieves batch auditing to support efficient handling of multiple auditing tasks. In 2011, based on fragment structure, random sampling and index-hash table, Zhu et al. [28] propose a dynamic audit service for verifying the integrity of an untrusted and outsourced storage, which supporting provable updates to outsourced data, and timely abnormal detection. Since then, many other auditing mechanisms such as [13, 14, 15, 16, 18, 19, 23, 24, 25, 26, 27] have been proposed for protecting the integrity of the data in the cloud.

In 2013, Wang boyang et al. [15] proposed a novel public auditing mechanism for the integrity of shared data with efficient user revocation. By utilizing proxy re-signatures, it allows the cloud to re-sign blocks on behalf of existing users during user revocation phase, so that existing users do not need to download and re-sign blocks by themselves. In addition, a public verifier is always able to audit the integrity of shared data without retrieving the entire data from the cloud, even if some part of shared data has been re-signed by the cloud. Wang [19] proposed the concept of proxy provable data possession (PPDP), which is a matter of crucial importance when the user can not perform the remote data possession checking. Based on bilinear pairings, he gives an efficient and provable secure PPDP protocol. In 2014, Wang et al. [20] firstly formalized the model of identity based remote data possession checking (ID-RDPC) protocol for secure cloud storage. Based on bilinear pairings, they proposed the first concrete ID-RDPC protocol which was proven secure under the CDH assumption.

In this paper, we study the above three auditing mechanisms for secure cloud storage, including public auditing mechanism with user revocation [15], proxy provable data possession [19] and identity based remote data possession checking [20]. We show that the three schemes do not satisfy the property of correctness. When the active adversaries can arbitrarily tamper the cloud data and produce a valid auditing response to pass the verification. Therefore, the adversaries can cheat the auditor to believe that the data in cloud are well-maintained while in fact the data have been modified. Then, we discuss the origin of the security flaws and give a solution.

2 Preliminaries

In this section, we briefly review the basic concepts on bilinear pairings and the related system models.

2.1 Bilinear Pairings

Let $\mathbb{G}_1$ and $\mathbb{G}_2$ be two multiplicative cyclic groups of prime order p and let g be a generator of $\mathbb{G}_1$. The map e: $\mathbb{G}_1 \times \mathbb{G}_1 \rightarrow \mathbb{G}_2$ is said to be an admissible bilinear pairing with the following properties:

1) **Bilinearity:** $e(u^a, v^b) = e(u, v)^{ab}$ for all $u, v \in \mathbb{G}_1$ and for all $a, b \in \mathbb{Z}_p$;

2) **Non-degeneracy:** $e(g, g) \neq 1_{\mathbb{G}_1}$;

3) **Computability:** There exists an efficient algorithm to compute $e(u, v)$ for all $u, v \in \mathbb{G}_1$.

We note the modified Weil and Tate pairings associated with supersingular elliptic curves are examples of such admissible pairings.

2.2 System Model

The system model of public auditing mechanism with user revocation can be shown in Figure 1 [15]. The three entities: the cloud, the third party auditor (TPA), and users are involved in public auditing mechanism. The cloud offers data storage and sharing services to users. The TPA is able to publicly audit the integrity of shared data in the cloud for users. In a group, there is one original user and a number of group users. The original user is the original owner of data. This original user creates and shares data with other users in the group through the cloud. Both the original user and group users are able to access, download and modify shared data. Shared data is further divided into a number of blocks. A user can modify a block in shared data by performing an insert, delete or update operation on the block.

The system model of proxy provable data possession (PPDP) can be shown in Figure 2 [19]. The PPDP system consists of three different entities: user, public cloud service (PCS), and proxy. The user moves the massive data to the remote PCS, the PCS has significant storage space and computation resource to maintain the users' data; the proxy, which is delegated to check user's data possession.

The system model of identity based remote data possession checking (ID-RDPC) can be shown in Figure 3 [20]. The ID-RDPC protocol consists of three different entities: private key generator (PKG), public cloud service (PCS) and client. The PKG generates the public parameters and master public key and client's private key. The client moves the data to be stored on the public cloud

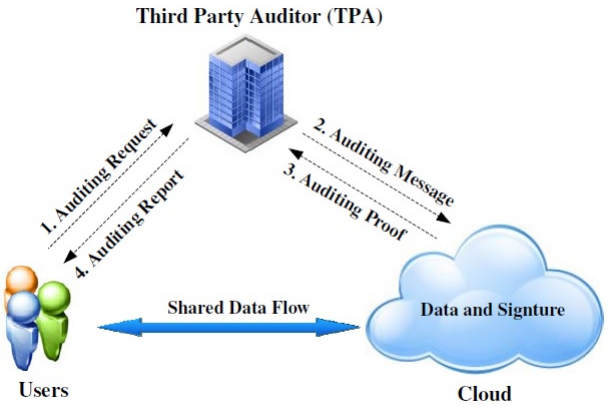

Figure 1: System model of public auditing mechanism

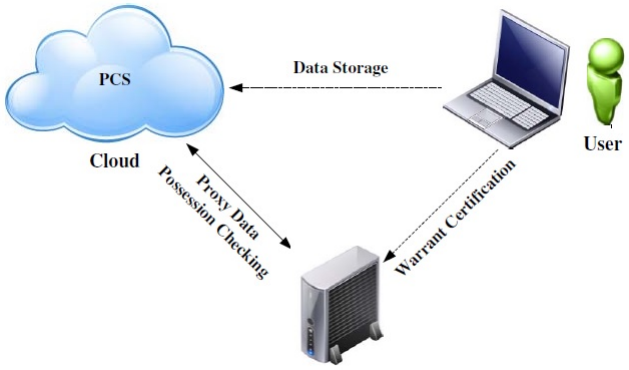

Figure 2: System model of PPDP

for maintenance and computation. The PCS has significant storage space and computation resource to maintain the users' data.

3 The Security of Three Auditing Mechanisms

In this section, we firstly review public auditing scheme with user revocation [15], proxy provable data possession [19] and identity based remote data possession checking [20], and then give the security analysis of three mechanisms in active adversaries.

3.1 Overview of Wang et al.'s Scheme

This scheme consists of six procedures: **KeyGen**, **ReKey**, **Sign**, **ReSign**, **ProofGen** and **ProofVerify**. The reader can refer to [15] for detailed description.

Let $\mathbb{G}_1$ and $\mathbb{G}_2$ be two groups of order p, g be a generator of $\mathbb{G}_1$, e: $\mathbb{G}_1 \times \mathbb{G}_1 \rightarrow \mathbb{G}_2$ be a bilinear map, ω be a random element of $\mathbb{G}_1$. The global parameters are $\{e, p, \mathbb{G}_1, \mathbb{G}_2, g, \omega, H, H'\}$, where H is a hash function with H: $\{0,1\}^* \rightarrow \mathbb{G}_1$ and H' is a hash function with H': $\{0,1\}^* \rightarrow \mathbb{Z}_p$. The total number of blocks in shared data is n, and shared data is described as $M = (m_1, \cdots, m_n)$.

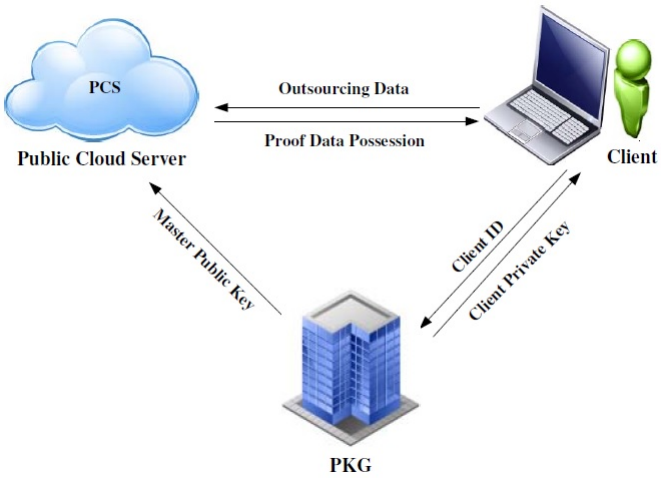

Figure 3: System model of ID-RDPC

The total number of users in the group is d.

KeyGen. For user u_i, he/she generates a random $x_i \in \mathbb{Z}_p$, and outputs his/her public key $pk_i = g^{x_i}$ and private key $sk_i = x_i$. Without loss of generality, we assume user u_1 is the original user, who is the creator of shared data. The original user also creates a user list (UL), which contains ids of all the users in the group. The user list is public and signed by the original user.

ReKey. The cloud generates a re-signing key $rk_{i\rightarrow j}$ as follows:

1) The cloud generates a random $r \in \mathbb{Z}_p$ and sends it to user u_i;
2) User u_i sends r/x_i to user u_j, where $sk_i = x_i$;
3) User u_j sends rx_j/x_i to the cloud, where $sk_j = x_j$;
4) The cloud recovers $rk_{i\rightarrow j} = x_j/x_i$.

Sign. Given private key $sk_i = x_i$, block $m_k \in \mathbb{Z}_p$ in shared data M and its block identifier id_k, where $k \in [1, n]$, the user u_i outputs the signature on block m_k as $\sigma_k = (H(id_k)\omega^{m_k})^{x_i}$.

ReSign. When user u_i is revoked from the group, the cloud is able to convert signatures of user u_i into signatures of user u_j on the same block. More specifically, given re-signing key $rk_{i\rightarrow j}$, public key pk_i, signature σ_k, block m_k and block identifier id_k, the cloud first checks that $e(\sigma_k, g) = e(H(id_k)\omega^{m_k}, pk_i)$. If the verification result is 0, the cloud outputs $\perp$; otherwise, it outputs $\sigma'_k = \sigma_k^{rk_{i\rightarrow j}} = (H(id_k)\omega^{m_k})^{x_i \cdot x_j/x_i} = (H(id_k)\omega^{m_k})^{x_j}$. After the re-signing, the original user removes user u_i's id from UL and signs the new UL.

ProofGen. To audit the integrity of shared data, the TPA generates an auditing message as follows:

1) Randomly picks a c-element subset L of set $[1, n]$ to locate the c selected random blocks that will be checked in this auditing task;
2) Generates a random $y_l \in \mathbb{Z}_p$, for $l \in L$ and q is a much smaller prime than p;
3) Outputs an auditing message $\{(l, y_l)\}_{l\in L}$, and sends it to the cloud.

After receiving an auditing message, the cloud generates a proof of possession of shared data M. More concretely,

1) The cloud divides set L into d subset $L_1, L_2, \cdots, L_d$, where L_i is the subset of selected blocks signed by user u_i. And the number of elements in subset L_i is c_i. Clearly, we have $c = \sum_{i=1}^{d} c_i$, $L = L_1 \cup \cdots \cup L_d$ and $L_i \cap L_j = \Phi$ for $i \neq j$;
2) For each set L_i, the cloud computes $\alpha_i = \sum_{l\in L_i} y_l m_l \in \mathbb{Z}_p$ and $\beta_i = \prod_{l\in L_i} \sigma_l^{y_l} \in \mathbb{G}_1$;
3) The cloud outputs an auditing proof $\{\alpha, \beta, \{id_l\}_{l\in L}\}$, and sends it to the verifier, where $\alpha = (\alpha_1, \cdots, \alpha_d)$ and $\beta = (\beta_1, \cdots, \beta_d)$.

ProofVerify. With an auditing proof $\{\alpha, \beta, \{id_l\}_{l\in L}\}$, an auditing message $\{(l, y_l)\}_{l\in L}$, and all the existing users'public keys $(pk_1, \cdots, pk_d)$, the TPA checks the correctness of this auditing proof as

$$e(\prod_{i=1}^{d} \beta_i, g) = \prod_{i=1}^{d} e(\prod_{l\in L_i} H(id_l)^{y_l} \cdot \omega^{\alpha_i}, pk_i).$$

If the result is 1, the verifier believes that the integrity of all the blocks in shared data M is correct. Otherwise, the verifier outputs 0.

3.2 Security Analysis on Wang et al.'s Scheme

As the audit scheme for shared data with efficient user revocation in cloud, Wang boyang et al.'s scheme enjoys many desirable security properties. Informally, this mechanism needs that it be infeasible to fool the TPA into accepting false statements, i.e. the TPA is able to correctly detect whether there is any corrupted data.

However, we show that when an active adversary, such as a bug planted in the software running on the cloud server by a malicious programmer or a hacker, is involved in the auditing process. Specifically, the adversary can arbitrarily modify or tamper the outsourced data and fool the TPA to believe the data are well preserved in the cloud.

All the information the adversary has to know is how the data are modified. The details are described as follows:

1) For each set L_i, the adversary $\mathcal{A}$ firstly modifies the block m_l to $m_l^* = m_l + f_l$ for $l \in L_i$ and records the values f_l, i.e. how the shared data are modified.
2) When getting the challenge $\{(l, y_l)\}_{l\in L}$ from the TPA, the cloud honestly executes **ProofGen** algorithm to compute $\alpha^* = (\alpha_1^*, \cdots, \alpha_d^*), \beta = (\beta_1, \cdots, \beta_d)$ as follows: For $l \in L_i$, the cloud computes

$$\begin{aligned}\alpha_i^* &= \sum\nolimits_{l\in L_i} y_l m_l^* = \sum\nolimits_{l\in L_i} y_l(m_l + f_l) \\ &= \alpha_i + \sum\nolimits_{l\in L_i} y_l f_l \\ \beta_i &= \prod\nolimits_{l\in L_i} \sigma_l^{y_l}.\end{aligned}$$

Finally, the cloud sends the auditing proof $(\alpha^*, \beta, \{id_l\}_{l\in L})$ to the TPA.

3) The adversary $\mathcal{A}$ intercepts the auditing proof $(\alpha^*, \beta, \{id_l\}_{l\in L})$ from the cloud to TPA, and modifies each α_i^* to $\alpha_i = \alpha_i^* - \sum_{l\in L_i} y_l f_l$ for $l \in L_i$.

By performing such a modification, the adversary $\mathcal{A}$ derives a correct proof with respect to the original data blocks M, and sends it to the TPA. As a result, in **ProofVerify** phase, the modified auditing proof can make the equation hold and, thus the TPA believes that shared data are all well-maintained, while the data have been polluted by the adversary $\mathcal{A}$.

3.3 Overview of Wang's Scheme

This scheme consists of six procedures: **SetUp**, **TagGen**, **SignVerify**, **CheckTag**, **GenProof** and **CheckProof**. The reader can refer to [19] for detailed description.

Suppose the maximum number of the stored block-tag pairs is n. Let f: $\mathbb{Z}_q^* \times \{1, 2, \cdots, n\} \to \mathbb{Z}_q^*$ and Ω: $\mathbb{Z}_q^* \times \{1, 2, \cdots, n\} \to \mathbb{Z}_q^*$ be two pseudo-random functions, and let π: $\mathbb{Z}_q^* \times \{1, 2, \cdots, n\} \to \{1, 2, \cdots, n\}$ be a pseudo-random permutation and H: $\mathbb{G}_2 \times \{0, 1\} \to \mathbb{Z}_q^*$, h: $\mathbb{Z}_q^* \to \mathbb{G}_1$ be cryptographic hash functions.

Let g be a generator of $\mathbb{G}_1$. We assume that the file F (maybe encoded by using error-correcting code, such as, Reed-Solomon code) is divided into n blocks $(m_1, m_2, \cdots, m_n)$ where $m_i \in \mathbb{Z}_q^*$ and q is the order of $\mathbb{G}_1$ and $\mathbb{G}_2$. Without loss of generality, we denote $F = (m_1, m_2, \cdots, m_n)$.

SetUp. The user picks a random number $x \in \mathbb{Z}_q^*$ as its private key and computes $X = g^x$ as its public key. The PCS picks a random number $y \in \mathbb{Z}_q^*$ as its private key and computes $Y = g^y$ as its public key. The proxy picks a random number $z \in \mathbb{Z}_q^*$ as its private key and computes $Z = g^z$ as its public key. The user picks a random element $u \in \mathbb{G}_1$ and a secure signature/verification algorithm pair $(SigGen, SignVerify)$. Finally, the system parameter is $param = \{\mathbb{G}_1, \mathbb{G}_2, e, f, \Omega, \pi, H, h, X, Y, Z, u, q, (SigGen, SignVerify)\}$.

TagGen. Given $F = (m_1, m_2, \cdots, m_n)$ and the warrant ω, the user generates the tag T_{m_i} of the block m_i as follows:

1) Client computes $t = H(e(Y,Z)^x, \omega)$, $W_i = \Omega_t(i)$;
2) Client computes $T_{m_i} = (h(W_i)u^{m_i})^x$ for $i \in [1,n]$;
3) Client computes the signature $Sign = SigGen_x(\omega)$ on the warrant ω.

Then the user sends the the block-Tag pairs collection $\{(T_{m_i}, m_i), i \in [1,n]\}$ and the warrant ω to the PCS. The PCS stores the block-tag pairs and the warrant ω. The user deletes the block-tag pairs $\{(T_{m_i}, m_i), i \in [1,n]\}$ from its local storage. At the same time, the user sends the warrant-signature pair $(\omega, Sign)$ to the proxy.

SignVerify. Upon receiving (ω, $Sign$), the proxy performs the verification algorithm $SignVerify(\omega, Sign, X)$. If it is valid, the proxy accepts this warrant ω; otherwise, the proxy rejects it and queries the user for new warrant-certification pair.

CheckTag. Given $\{(T_{m_i}, m_i), i \in [1,n]\}$, PCS performs the procedures for every $i, 1 \le i \le n$, as follows:

1) PCS computes $\hat{t} = H(e(X,Z)^y, \omega)$ and $\hat{W}_i = \Omega_{\hat{t}}(i)$;
2) PCS verifies whether $e(T_i, g) = e(h(\hat{W}_i)u^{m_i}, X)$ holds.

If it holds, then PCS accepts it. Otherwise, PCS rejects it and queries the user for new block-tag pair.

GenProof. Let the challenge be $chal = (c, k_1, k_2)$ where $1 \le c \le n$, $k_1 \in \mathbb{Z}_q^*$, $k_2 \in \mathbb{Z}_q^*$. In this phase, the proxy asks the PCS for remote data possession proof of c file blocks whose indexes are randomly chosen using a pseudo-random permutation keyed with a fresh randomly chosen key for each challenge. On the other hand, the proxy sends $(\omega, Sign)$ to PCS. PCS verifies whether the signature $Sign$ is valid. If it is valid, PCS compares this ω with its stored warrant ω'. When $\omega = \omega'$ and the proxy's queries comply with the warrant ω, PCS performs the procedures as follows. Otherwise, PCS rejects the proxy's query.

1) For $1 \le j \le c$, PCS computes the indexes and coefficients of the blocks for which the proof is generated: $i_j = \pi_{k_1}(j)$, $a_j = f_{k_2}(j)$;
2) PCS computes $T = \prod_{j=1}^{c} T_{m_i}^{a_j}$, $\hat{m} = \sum_{j=1}^{c} a_j m_{i_j}$.

PCS outputs $V = (\hat{m}, T)$ and sends V to the proxy as the response.

CheckProof. Upon receiving the response V from PCS, the proxy performs the procedures as follows:

1) Proxy computes $t = H(e(X,Y)^z, \omega)$;
2) Proxy checks whether the following formula holds

$$e(T,g) = e(\prod_{i=1}^{c} h(\Omega_t(\pi_{k_1}(i)))^{f_{k_2}(i)} u^{\hat{m}}, X).$$

If it holds, then the proxy outputs "success". Otherwise the proxy outputs "failure".

3.4 Security Analysis on Wang's Scheme

As the proxy provable data possession (PPDP) in public cloud, Wang's scheme enjoys many desirable security properties of a PPDP scheme including correctness and unforgeability. Informally, correctness property means that it be infeasible to fool the proxy into accepting false statements, i.e. the proxy is able to correctly detect whether there is any corrupted data.

Similar to the analysis of Wang boyang et al.'s scheme, we show that the Wang's scheme is not secure in active adversary, who can arbitrarily modify or tamper the outsourced data and fool the proxy to believe the data are well preserved without being detected by the proxy in the cloud. The details are described as follows:

1) The adversary $\mathcal{A}$ firstly modifies the block $m_{i,j}$ to $m_{i,j}^* = m_{i,j} + d_{i,j}$ for $i \in [1,n]$, $j \in [1,c]$ and records the values $d_{i,j}$.
2) When PCS receiving the challenge $chal = (c, k_1, k_2)$ from the proxy, the PCS honestly executes **GenProof** algorithm to compute $V^* = (\hat{m}^*, T)$ as follows:

$$\begin{aligned} \hat{m}^* &= \sum_{j=1}^{c} a_j m_{i_j}^* = \sum_{j=1}^{c} a_j (m_{i,j} + d_{i,j}) \\ &= \hat{m} + \sum_{j=1}^{c} a_j d_{i,j} \\ T &= \prod_{j=1}^{c} T_{m_{i_j}}^{a_j}. \end{aligned}$$

Then it sends the $V^* = (\hat{m}^*, T)$ auditing proof to the proxy.

The adversary $\mathcal{A}$ intercepts the auditing proof $V^* = (\hat{m}^*, T)$ and modifies $\hat{m}^*$ to $\hat{m} = \hat{m}^* - \sum_{j=1}^{c} a_j d_{i,j}$. By performing such a modification, the adversary $\mathcal{A}$ obtains a correct proof with respect to the original data $m_{i,j}$ for $i \in [1,n]$ and $j \in [1,c]$. As a result, the proof can pass the auditing verification, which makes the proxy believe that the shared data are well maintained by the PCS, while in fact the data have been corrupted.

3.5 Overview of Wang et al.'s Scheme

An ID-RDPC protocol is a collection of five polynomial-time algorithms: **Setup**, **Extract**, **TagGen**, **GenProof** and **CheckProof**. The reader can refer to [20] for detailed description.

Setup. PKG chooses a random number $x \in \mathbb{Z}_q^*$ and sets $Y = g^x$, where g is a generator of the group $\mathbb{G}_1$. PKG chooses a random element $u \in \mathbb{G}_1$. Define two cryptographic hash functions: H: $\{0,1\} \to \mathbb{Z}_q^*$, h: $\mathbb{Z}_q^* \to \mathbb{G}_1$. Let f be a pseudo-random function and let π be a pseudo-random permutation f: $\mathbb{Z}_q^* \times \{1,2,\cdots,n\} \to$

$\mathbb{Z}_q^*$, π: $\mathbb{Z}_q^* \times \{1, 2, \cdots, n\} \to \{1, 2, \cdots, n\}$. PKG publishes $(\mathbb{G}_1, \mathbb{G}_2, e, q, g, Y, u, H, h, f, \pi)$ and keeps x as the master key.

Extract. A client submits the identity ID to the PKG. The PKG picks $r \in \mathbb{Z}_q^*$ and computes $R = g^r, \sigma = r + xH(ID, R) \bmod q$. The PKG sends the private key $sk_{ID} = (R, \sigma)$ to the client by a secure channel. The client can verify the correctness of the received private key by checking whether $g^\sigma = R \cdot Y^{H(ID,R)}$ holds. If the previous equality holds, the client accepts the private key; otherwise, he/she rejects it.

TagGen. We assume that the client generates the tags sequentially according to the counter i. That is, the client generates a tag for a message block m_2 after m_1, which implies that the client maintains the latest value of the counter i. For m_i, the client performs the TagGen procedure as follows:

1) Compute $T_i = ((h(i)u^{m_i})^\sigma$;
2) Output T_i and send to the PCS.

GenProof. In this phase, the verifier (who can be the client himself/herself) queries the PCS for a proof of data possession of c file blocks whose indices are randomly chosen using a pseudo-random permutation keyed with a fresh random-chosen key for each challenge.

The number $k_1 \in Z_q^*$ is the random key of the pseudo-random permutation π. Also, $k_2 \in Z_q^*$ is the random key of the pseudo-random function f. Let the challenge be $chal = (c, k_1, k_2)$. Then, the PCS does:

1) For $1 \leqslant j \leqslant c$, compute the indices and coefficients of the blocks for which the proof is generated: $i_j = \pi_{k_1}(j)$, $a_j = f_{k_2}(j)$. In this step, the challenge $chal$ defines an ordered set $\{c, i_1, \cdots, i_c, a_1, \cdots, a_c\}$;
2) Compute $T = \prod_{j=1}^{c} T_{i_j}^{a_j}$, $\hat{m} = \sum_{j=1}^{c} a_j m_{i_j}$;
3) Output $V = (T, \hat{m})$ and send V to the client as the response to the $chal$ query.

CheckProof. Upon receiving the response V from the PCS, the verifier (who can be the client himself/herself) does:

1) Check the equation: $e(T, g) = e\left(\prod_{i=1}^{c} h\left(\pi_{K_1}(i)\right)^{f_{k_2}(i)} u^{\hat{m}}, R \cdot Y^{H(ID,R)}\right)$.
2) If the previous equation holds, output "success". Otherwise, output "failure".

3.6 Security Analysis on Wang et al.'s Scheme

As an Identity based remote data possession checking (ID-RDPC) protocol in the public clouds, Wang et al.'s scheme [20] enjoys many desirable security properties. Informally, this mechanism needs that it be infeasible to fool the client into accepting false statements, i.e. the client is able to correctly detect whether there is any corrupted data.

However, we show that when an active adversary, such as a bug planted in the software running on the cloud server by a malicious programmer or a hacker, is involved in the remote data possession checking process. Specifically, the active adversary can arbitrarily modify or tamper the outsourced data and fool the PCS to believe the data are well preserved in the cloud.

All the information the adversary has to know is how the data are modified. The details are described as follows:

1) The adversary $\mathcal{A}$ firstly modifies the block m_{i_j} to $m_{i_j}^* = m_{i_j} + d_{i_j}$ for $j \in [1, c]$ and records the values d_{i_j}.
2) When the PCS receiving the challenge $chal = (c, k_1, k_2)$ from the client, the PCS honestly executes **GenProof** algorithm to compute $V^* = (\hat{m}^*, T)$ as follows:

$$\begin{aligned} \hat{m}^* &= \sum\nolimits_{j=1}^{c} a_j m_{i_j}^* = \sum\nolimits_{j=1}^{c} a_j (m_{i,j} + d_{i,j}) \\ &= \hat{m} + \sum\nolimits_{j=1}^{c} a_j d_{i,j} \\ T &= \prod\nolimits_{j=1}^{c} T_{m_{i_j}}^{a_j}. \end{aligned}$$

Then it sends the response $V^* = (\hat{m}^*, T)$ to the client.

The adversary $\mathcal{A}$ intercepts the response $V^* = (\hat{m}^*, T)$ and modifies $\hat{m}^*$ to $\hat{m} = \hat{m}^* - \sum_{j=1}^{c} a_j d_{i_j}$. By performing such a modification, the adversary $\mathcal{A}$ obtains a correct proof with respect to the original data m_{i_j} for $j \in [1, c]$. As a result, the proof can pass the verification, which makes the client believe that the shared data are well maintained by the PCS, while in fact the data have been corrupted.

3.7 Discussion on the Origin and Method

Through the above analysis, it is known Wang boyang et al.'s scheme, Wang's scheme and Wang et al.'s scheme can not satisfy the correctness. Taking use of our proposed attack methods, an active adversary can arbitrarily modify or tamper the outsourced data in the cloud and fool the auditor (TPA, Proxy and client) to believe the data are well preserved in the cloud.

Why does the above three schemes have security flaw? There is no an authentication check on the auditing response about the challenge. This property incurs the security flaws. It is important to clarity the security in order to design secure and practical auditing mechanism in cloud storage.

To solve the security problem, we can use the technique of digital signature. Specifically, in auditing response, the cloud service firstly uses the private key to compute a

signature for proof and, then send both the proof and the corresponding signature as the response to the challenge. Receiving the response, the auditor first verifies whether the signature is valid or not. If it is valid, the auditor performs the auditing verification protocol; otherwise, the auditor discards the response. Therefore, if the shared data have been modified, the auditor must be able to detect it because of the employment of the digital signatures.

We can use HMAC technique [8], which also provided message authentication function. Meantime, the 3-move protocol [25, 26, 27]: commitment, challenge and response (for example Schnorr's protocol [10]) is also used in auditing response and prevent to the pollution attacks from active adversaries.

4 Conclusion

With the development of cloud computing, many people focused on the study of information security in Cloud Environments [5, 9, 21]. Wang boyang et al. [15], Wang [19] and Wang et al. [20] proposed a secure auditing mechanisms for the stored data in cloud. However, through cryptanalysis, we show that their schemes still has security weakness. By giving a concrete attacks, we prove that the two schemes are not secure in active adversary environment. Specially, any adversary can modify the data without being detected by the auditor in the verification phase. Finally, we study the origin of the security flaw and propose a solution to remedy this weakness.

Acknowledgments

This work was supposed by the National Natural Science Foundation of China (No.61202438), the China Postdoctoral Science Foundation (No.2011M501427), the Key Project of Industry Science and Technology of Shaanxi Province (No.2015GY014) and the Project of Technology Transfer Promoting Engineering of Xi'an City (No.CXY1437(10)).

References

[1] M. Armbrust, et al, "A view of cloud computing," *ACM Communication*, vol. 53, pp. 50–58, 2010.

[2] G. Ateniese, R. C. Burns, R. Curtmola, J. Herring, L. Kissner, Z. N. J. Peterson, and D. X. Song, "Provable data possession at untrusted stores," in *Proceedings of the 2007 ACM Conference on Computer and Communications Security (CCS'07)*, pp. 598–609, 2007.

[3] G. Ateniese, R. D. Pietro, L. V. Mancini, G. Tsudik, "Scalable and efficient provable data possession," in *Proceedings of the 4th ACM International Conference on Security and Privacy in Communication Networks (SecureComm'08)*, pp. 1–10, 2008.

[4] D. Boneh, B. Lynn, H. Shacham, "Short signatures from the Weil pairing," *Journal of Cryptology*, vol. 17, no. 4, pp. 297–319, 2004.

[5] P. S. Chung, C. W. Liu and M. S. Hwang, "A study of attribute-based proxy re-encryption scheme in cloud environments," *International Journal of Network Security*,vol. 16, no. 1, pp. 1–13, 2014.

[6] C. C. Erway, A. Kpc, C. Papamanthou, R. Tamassia, "Dynamic provable data possession," in *Proceedings of the 2009 ACM Conference on Computer and Communications Security (CCS'09)*, pp. 213–222, 2009.

[7] A. Juels and B. S. K. Jr, "PORS: Proofs of retrievability for large files," in *Proceedings of the 2007 ACM Conference on Computer and Communications Security (CCS'07)*, pp. 584–597, 2007.

[8] H. Krawczyk, M. Bellare and R. Canetti, *HMAC: Keyed-hashing for Message Authentication*, RFC 2104, Feb. 1997.

[9] C. C. Lee, P. S. Chung and M. S. Hwang, "A survey on attribute-based encryption schemes of access control in cloud environments," *International Journal of Network Security*, vol. 15, no. 4, pp. 231–240, 2013.

[10] C. P. Schnorr, "Efficient identification and signatures for smart cards," in *Advances in Cryptology (Crypto'89)*, LNCS 435, pp. 239–252, 1990.

[11] H. Shacham, B. Waters, "Compact proofs of retrievability," in *Advances in Cryptology (Asiacrypt'08)*, LNCS 5350, pp. 90–107, 2008.

[12] L. M. Vaquero, L. Rodero-Merino, J. Caceres, M. Lindner, "A break in the clouds: Towards a cloud definition," *ACM SIGCOMM Computer Communication Review*, vol. 39, pp. 50–55, 2009.

[13] B. Wang, B. Li, H. Li, "ORUTA: Privacy-preserving public auditing for shared data in the cloud," in *Proceedings of the IEEE International Conference on Cloud Computing*, pp. 293–302, 2012.

[14] B. Wang, B. Li, H. Li, "KNOX: Privacy-preserving auditing for shared data with large groups in the cloud," in *Applied Cryptography and Network Security (ACNS'12)*, LNCS 7341, pp. 507–525, 2012.

[15] B. Wang, B. Li and H. Li, "Public auditing for shared data with efficient user revocation in the cloud," in *Proceedings of IEEE INFOCOM'13*, pp. 2904–2912, 2013.

[16] C. Wang, K. Ren, W. Lou, J. Li, "Toward publicly auditable secure cloud data storage services," *IEEE Networks*, vol. 24, no. 4, pp. 19–24, 2010.

[17] C. Wang, Q. Wang, K. Ren, W. Lou, "Privacy-preserving public auditing for data storage security in cloud computing," in *Proceedings IEEE of INFOCOM'10*, pp. 1–9, 2010.

[18] C. Wang, Q. Wang, K. Ren, N. Cao, W. Lou, "Toward secure and dependable storage services in cloud computing," *IEEE Transactions on Services Computing*, vol. 5, no. 2, pp. 220–232, 2012.

[19] H. Wang, "Proxy provable data possession in public clouds," *IEEE Transactions on Services Computing*, vol. 6, no. 4, pp. 551–559, 2013.

[20] H. Wang, Q. Wu, B. Qin and J. Domingo-Ferrer, "Identity-based remote data possession checking in public clouds," *IET Information Security*, vol. 8, no. 2, pp. 114–121, 2014.

[21] J. Wang, X. Yu and M. Zhao, "Fault-tolerant verifiable keyword symmetric searchable encryption in hybrid cloud," *International Journal of Network Security*, vol. 17, no. 4, pp. 471–483, 2015.

[22] Q. Wang, C. Wang, J. Li, K. Ren, W. Lou, "Enabling public verifiability and data dynamics for storage security in cloud computing," in *Proceedings of the 14th European Symposium on Research in Computer Security (ESORICS'09)*, LNCS 5789, pp. 355–370, 2009.

[23] Q. Wang, C. Wang, K. Ren, W. Lou, J. Li, "Enabling public auditability and data dynamics for storage security in cloud computing," *IEEE Transactions on Parallel Distribute System*, vol. 22, no. 5, pp. 847–859, 2011.

[24] K. Yang, X. Jia, "Data storage auditing service in cloud computing: Challenges, methods and opportunities," *World Wide Web*, vol. 15, no. 4, pp. 409–428, 2012.

[25] Y. Zhu, G. J. Ahn, H. Hu, S. S. Yau, Ho G. An and C. J. Hu, "Dynamic audit services for outsourced storages in Clouds," *IEEE Transactions on Services Computing*, vol. 6, no. 2, pp. 227–238, 2013.

[26] Y. Zhu, H. Hu, G. J. Ahn, S. S. Yau, "Efficient audit service outsourcing for data integrity in clouds", *The Journal of Systems and Software*, vol. 85, pp. 1083–1095, 2012.

[27] Y. Zhu, H. Hu, G. J. Ahn, M. Yu, "Cooperative provable data possession for integrity verification in multi-cloud storage," *IEEE Transactions on Parallel Distribute System*, vol. 23, no. 12, pp. 2231–2244, 2012.

[28] Y. Zhu, H. Wang, Z. Hu, G.-J. Ahn, H. Hu, S.S. Yau, "Dynamic audit services for integrity verification of outsourced storages in clouds," in *Proceedings of the 2011 ACM Symposium on Applied Computing (SAC'11)*, pp. 1550–1557, 2011.

Yang Ming was born in Shaanxi Province, China in 1979. He received the B.S. and M.S. degrees in mathematics from Xian University of Technology in 2002 and 2005 respectively, and the Ph.D. degree in cryptography from Xidian University in 2008. Currently he is a supervisor of postgraduate and associate professor of Chang'an University. His research interests include cryptography and digital signature.

Yumin Wang was born in Beijing, China in 1936. He received the B.S. degree from the Department of Telecommunication Engineering, Xidian University in 1959. In 1979-1981, he was a visiting scholar in Department of Electronic Engineering, Hawaii University. Currently he is a doctoral supervisor and professor of Xidian University. He is a fellow member of the Board of Governors of the Chinese Institute of Cryptography (preparatory committee) and a Senior Member (SM) of IEEE. His research interests include information theory, coding, and cryptography.

Reviewers (Volume 17, 2015)

Ahmed Abd El-Rahiem Abd El-Latif
Maha Abdelhaq
Ahmed Mohammed Abdullah
Ashwini B Abhale
Wael Adi
Asimi Ahmed
Shahid Alam
Ruhul Amin
Rengarajan Amirtharajan
Benjamin Arazi
Razi Arshad
Shashikant V Athawale
Felix Au-Yeung
Karthikeyan B
Parameshachari B D
Ayman Bahaa
Nazrulazhar Bahaman
Hatem M. Bahig
Wang Bailing
Pankaj Bajaj
Siles Balasingh
Kavitha Balu
Pijush Barthakur
Mohammad Beheshti Atashgah
Li Bin
Andrew Blyth
Mitko Bogdanoski
Zhengjun Cao
Tianjie Cao
Nilotpal Chakraborty
Chi-Shiang Chan
Chin-Chen Chang
Jan Min Chen
Zhenhua Chen
Tzung-Her Chen
Xiangguo Cheng
Qingfeng Cheng
Lin Cheng
Hu Chengyu
Aayush Chhabra
Jue-Sam Chou
Patrick G Clark
Nicolae Constantin
Prodipto Das
Ashok Kumar Das
Angsuman Das
Subhasish Dhal
Dhanashri Devendra Dhokate
Nizamud Din
Nishant Doshi
Tarun Dubey
Fadi El-Hassan
Ziba Eslami
Tung Huang Feng
Amjad Gawanmeh
Faiq Gmira
Guang Gong
Partha Sarathi Goswami
Kumar Gunjan
Rui Guo
Swati Gupta
Chien-Lung Hsu
Chin-Tser Huang
Phoenix Huang
Huajun Huang
Md Ruhul Islam
Aws Naser Jaber
Pradip M Jawandhiya
Vijay Kumar Jha
Tsai Jia Lun
Qi Jiang
Shaoquan Jiang
Lin Zhi Jiang
Ashish Joshi
Nirmalya Kar
Rasool Kazemi
Liam Keliher
Asia Samreen Khan
Brian King
Naresh N Kumar
Shyam Nandan Kumar
Saru Kumari
Nishchal Kush
Duong Hai Le
Chun-Ta Li
Keying Li
Jiguo Li
Tian You Liang
Kriangkrai Limthong
Chia-Chen Lin
Rongxing Lu
Hemalatha M
Huan Ma
Rajiv Mahajan
William R Mahoney
Tanmoy Maitra
Mohd Zaki Mas'ud
Ali Mirarab
Nikolay Andreevich Moldovyan
Guillermo Morales-Luna
Hamdy M. Mousa
Kuntal Mukherjee
Zulkiflee Muslim
Amitava Nag

Preeti Nagrath
Kanagaraj Narayanasamy
Vankamamidi Srinivasa Naresh
Prabir Kr Naskar
Siaw-Lynn Ng
Xuyun Nie
Uzoma Emmanuel Opara
Shengyi Pan
Arun Raj Kumar Parthiban
Bhautik Kishorbhai Patel
Pravin Kalgonda Patil
Hoi Ting Poon
Sanjeev Puri
Murad Abdo Rassam Qasm
Chuan Qin
Subha Lakshmi R T
Y. Sreenivasa Rao
Rama Chandra Rao
Prabhudutta Ray
M. Gnaneswara Reddy
Yanli Ren
Habib Rostami
Sandhya S
Yassine Sadqi
Maryam Saeed
Hemlal Sahu
Hemraj Saini
Mahmoud Salmasizad
Sabyasachi Samanta
Ashutosh Saxena
Michael Scott
Resmi Sekhar
Elena Sendroiu
Seyyed Amin Seyyedi
Aamir Shahzad
Shahaboddin Shamshirband
Uvaraj Shan
Xiao Shangqin
Tarun Narayan Shankar
Gaurav Sharma
Hamdy S Soliman
Makam Venkata Subramanyam
Yinxia Sun
Weiqing Sun
Nedal Mohammad Tahat
Zuowen Tan
Ciza Thomas
Tony Thomas
Miaomiao Tian
Rajesh Kumar Tiwari
Geetam Singh Tomar
Yuan-Yu Tsai
S. C. Tsaur
Mangesh Rameshrao Umak
Subba Rao Y V
Hemanth Varanasi
Pallapa Venkataram
Vandani Verma
S. Maria Celestin Vigila
Feng Wang
Dazhong Wang
Daxing Wang
Ying Wang
Yongtao Wang
Shuozhong Wang
Lei Xu
Degang Xu
Benlcouiri Y
Xu Yan
Li Yang
Wei Yang
Li Yanjun
Ali Adel Yassin
Venkatramana Reddy Yeddula
Huang Yiwang
Milad Yousefi
Huifang Yu
Yong Zeng
Jianping Zeng
Zhao Zhang
Xiaojun Zhang
Mingwu Zhang
Yinghui Zhang
Jie Xiu Zhang
Fangguo Zhang
Ze-Mao Zhao
Dong Zheng
Hongfeng Zhu
Frank Zhu

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.