IOSR Journal of Computer Engineering (IOSR-JCE)
e-ISSN: 2278-0661,p-ISSN: 2278-8727, Volume 16, Issue 5, Ver. VII (Sep – Oct. 2014), PP 50-55
www.iosrjournals.org

Improved Data Sharing Security with Dual-MAC Address Verification and SHCS

Arsha S Vasanthan
(Dept. of Computer Science and Engineering, Mahatma Gandhi University, Kerala, India)

***Abstract*:** *For the easy and secure sharing of an unknown private data, an algorithm is developed. The proposed algorithm has improved the data sharing security with MAC address and SHCS verifications. This technique enables the sender to effectively assign a value for the commit phase.MAC address verification improved the security of authorized users at both the sender and receiver side. At the receiver side, the commit value verification is done. These features have provided an increased privacy preservation of the shared data. The usage of SHCS and MAC address in this new technique prevents hackers during the passage of private data. It is only possible in this new algorithm that more complex data can be shared with maximum security. My thesis examines the latest scheme in the literature, AIDA algorithm and proposes a new algorithm for improving the data sharing security. The new algorithm is created on top of SHCS algorithm with the help of MAC address verification. XOR encryption and decryption technique is used in my work. A Base64 encoder and decoder mechanism is added along with the encryption and decryption technique.*

***Keywords*:** *MAC Verification and SHCS Verification, Anonymization and Deanonymization, Privacy Preservation and Data sharing security, XOR encryption and decryption, Encoder and Decoder.*

I. Introduction

Nowadays, internet has got more popularity as a communication medium. The internet popularity whether for personal or business use depends upon its anonymous communication support and data security. Businesses have their own reasons to engage in anonymous communication. The usage of anonymous communication avoids the consequences regarding the identity revelation and thereby the privacy is preserved. Researchers have been extensively worked to solve the problems caused by the sharing of private data among N parties. They have also done investigation in finding the relevance of anonymity, privacy preservation of nodes that share a private data and data sharing security level in various application domains: patient medical records [1], electronic voting [2], e-mail [3], social networking [4], etc.

All the above mentioned domains were successful in two topics. They are anonymity and privacy preservation. But the data sharing security was not up to the rank. Privacy preservation and anonymity can be implemented using AIDA algorithm [5], which is the existing system of my thesis work. AIDA algorithm posses less security during data sharing. My thesis work examines the above mentioned problem and proposes a new algorithm that ensures high security during data sharing. The new algorithm is build on top of MAC address verification and SHCS [6]. XOR Encryption and Decryption technique is used in the new algorithm. New algorithm holds the relevance of privacy preservation of nodes that share a private data and above all huge data sharing security is ensured. MAC address verification and SHCS algorithms played the main role in my thesis work in preventing the hackers. Hackers are the unauthorized persons who capture the shared data from various application domains. I have used MAC address verification in preventing the hackers.

Commitment schemes and their possible applications can be demonstrated with a simple example: Say that two people want to play rock-paper-scissors by email. The problem with trying to do so, is that one player may simply wait until they receive the other's email of, say, "rock" and then quickly reply with, say, "paper", winning the game. This problem can be overcome by commitment schemes. At the beginning of the game, each player commits to rock, paper, or scissors. After they have done so, each reveals the choice that they committed to earlier. It is not possible to cheat because as mentioned, commitment schemes are binding.

Interactions in a commitment scheme take place in two phases. The two phases are named as commit phase and reveal phase.

II. Survey on Literature Works

2.1 Secure Sum Algorithm

Secure Sum algorithm helps to share data by preserving the privacy of nodes that is going to share the data [5]. This algorithm was first explained by Adi Sharma et al. [7]. His work shows how to divide data D into n pieces in such a way that D is easily reconstructable from any k pieces, but even complete knowledge of k - 1 pieces reveals absolutely no information about D. Here each division of actual data D is transmitted using Secure Sum.

Table I Transmission By A Secure Sum Execution

Nodes	$r_{i,1}$	$r_{i,2}$	$r_{i,3}$	$r_{i,4}$	d_i
$n_{i=1}$	13	-10	6	-3	**6**
$n_{i=2}$	7	3	-5	5	**10**
$n_{i=3}$	-8	11	12	-9	**6**
$n_{i=4}$	6	-8	-5	9	**2**
s_i=	**18**	**-4**	**8**	**2**	T = **24**

Algorithm: Say nodes $n_1,\ldots,n_N$ are given. Each holding a data item d_i wants to share $T = \sum d_i$ among the nodes without revealing the d_i values.

1) Each of the node n_i, $i = 1,\ldots,N$, selects random values $r_{i,1},\ldots,r_{i,N}$ such that
$$r_{i,1} + \ldots + r_{i,N} = d_i$$
2) Each one of the "random" value $r_{i,j}$ is transmitted from node n_i to node n_j . The sum of all these random numbers $r_{i,j}$ is, of course, the desired total T.
3) Each of the node n_j totals all the random values received as:
$$s_j = r_{1,j} + \ldots + r_{N,j}$$
4) Finally, each node n_i simply broadcasts s_i to all other nodes so that each node can compute:
$$T = s_1 + \ldots + s_N$$

Example: In Table I, one example of Secure Sum is shown. The data item values held by nodes n_1, n_2, n_3 and n_4 are $d_1 = 6$, $d_2 = 10$, $d_3 = 6$ and $d_4 = 2$ respectively. For example, node n_3 would transmit -8, 11, 12, and -9 to nodes n_1, n_2, n_3 (itself) and n_4 respectively. Node n_3 would receive 6, -5, 12, and -5 from nodes n_1, n_2, n_3 (itself) and n_4 respectively. Then node n_3 would compute and transmit the total $s_3 = 8$ of the values received to all nodes. Finally, n_3 would compute the total of all the second round transmissions received, 24 = 18 + -4 + 8 + 2.

Disadvantage: Secure Sum can only share the calculation value of the data items and not the data item itself.

2.2 Power Sums Algorithm

Power Sums algorithm solved the problem of Secure Sum. Power Sums algorithm helps to share the exact data value and Power Sums algorithm assured privacy preserved data sharing [5]. This algorithm was explained by Qi Xie and Urs Hengartner et al. [4]. Power Sums algorithm is created on top of Secure Sum algorithm.

Algorithm: Say nodes $n_1,\ldots,n_N$ are given. Each one of them holding a data item d_i .

1) Each of the node n_i computes d_i^n for n = 1, 2, …, N. The nodes then use the algorithm, secure sum to share knowledge of the power sums: i = 1, …, N.

$P_1 = \sum d_i^1$	$P_2 = \sum d_i^2$	…	$P_N = \sum d_i^N$

2) The power sums P_1, …, P_N are used to generate a polynomial which has d_1, …, d_N as its roots. Representing the Newton polynomial as:
$$p(x) = c_N x^N + \ldots + c_1 x + c_0 \qquad (1)$$
the values c_0, …, c_N are obtained from the equations:

$$c_N = -1$$
$$c_{N-1} = -1/1(c_N P_1)$$
$$c_{N-2} = -1/2(c_{N-1} P_1 + c_N P_2)$$
$$c_{N-m} = -1/m\sum(c_{N-m+k}P_k) \text{ where } k= 1, \ldots, m. \qquad (2)$$

3) Finally, the polynomial p(x) is solved by each node, to determine the roots d_1, …, d_N.

Table II Powers Of D_i, Chosen By Each Node Modulo P = 11

d_i^e	e = 1	e = 2	e = 3	e = 4
$n_{i=1}$	6	3	7	9
$n_{i=2}$	10	1	10	1
$n_{i=3}$	6	3	7	9
$n_{i=4}$	2	4	8	5
$\sum d_i^e$	$P_1 = 2$	$P_2 = 0$	$P_3 = 10$	$P_4 = 2$

Example: Say N = 4 nodes n_i want to share a data item d_i. The values are $d_1 = 6$ for n_1, $d_2 = 10$ for n_2, $d_3 = 6$ for n_3, and $d_4 = 2$ for n_4. Choice of the prime P = 11 will serve to represent these numbers. The modulus 11 inverses needed will be 1/2 = 6, 1/3 = 4, and 1/4 = 3. The nodes compute the power sums shown in Table II. Solving each of the Newton identities (2) in turn yields $c_4 = -1 = 10$, $c_3 = 2$, $c_2 = 9$, $c_1 = 1$ and $c_0 = 6$ and thus the polynomial of (1) is:
$$p(x) = 10x^4 + 2x^3 + 9x^2 + 1x + 6 \qquad (\text{mod } P = 11)$$

P_1, P_2, P_3, and P_4 values are received by all the nodes and can compute the polynomial and its roots to recover the original data items, 2, 6, 6, and 10, but not their indices.

Disadvantage: The Power Sums algorithm can share the data item itself but the data item should always be in the form of a number. Complex data cannot be shared using Power Sums.

2.3 AIDA Algorithm

AIDA is the best algorithm in preserving the privacy of the nodes that share the private data [5]. AIDA is an effective algorithm for assigning identifiers (IDs) to the nodes of a network in such a way that the IDs are anonymous. Given N nodes, a permutation of integers {1,…,N} is used for this assignment. The ID being assigned is known only to the respective nodes in the group. This algorithm is based on a method for anonymously sharing simple data and finally resulted for sharing of complex data. AIDA is build on top of Secure Sum and power Sum algorithms. The disadvantage of Power Sums was solved by AIDA algorithm.

Disadvantage: AIDA algorithm posses less security during data sharing.

III. New Algorithm with High Security

My Thesis work examines the above mentioned problem and proposes a new algorithm that ensures high security during data sharing. The new algorithm is build on top of MAC address [8] and SHCS.

3.1 Environment

A confidential web application say Indian Army was chosen for building my thesis. Though it is a web application, the login activity (both admin login and officer login) is restricted with in the army office. Each of the new officers can register with the confidential website through the system to which they are assigned. It is only after the approval of the admin that the officers can perform some action. Data sharing action of an authorized officer with other authorized officers is focused in my work. My thesis shows both the AIDA and New Algorithm working. During the approval phase, the admin stores the MAC address of the new system in the database through the generate MAC link in the confidential web application. This generated MAC address is used for verification during the data sharing process among a group of officers. XOR encryption and decryption is used in my work. Base64 encoder and decoder are added with the XOR encryption and decryption. The MAC address is one of the important factors in preventing the hackers from attacking the data sharing system.

3.2 Detailed AIDA Algorithm

In my thesis, initially a detailed version of AIDA algorithm is implemented. This section is fully used for the comparison of security parameter among the two algorithms. The message and inbox links in the Indian Army web application is created for AIDA algorithm implementation.

Algorithm: Say nodes n_1, …, n_N are given. At one stage, random numbers between 1 to S are selected by each of the node in the group, where S >= N.

1) Initially, set the number of assigned nodes to 0. Say A = 0.
2) Each one of the unassigned nodes, n_i selects a random number r_i in the range 1 to S. A previously assigned node selects 0 as its random number, i.e. $r_i = 0$.
3) The chosen random numbers are shared anonymously. The shared numbers are then denoted as q_1, …, q_N.
4) Let q_1, …, q_k represent a revised list of shared values. In this list, all the zero values and duplicated values are entirely removed where k is the number of unique random values. The nodes n_i that hold unique random numbers then determine their index s_i. It is determined from the position of their random numbers in the revised list as it would appear after being sorted:
$$s_i = A + \text{Card}\{q_j : q_j <= r_i\}$$
5) Next, update the number of assigned nodes: A = A + k.
6) If A < N, then go to step (2).

Table III Detailed Aida Algorithm Execution

R Step	A	r_1 r_2 r_3 r_4	q_1 q_2 q_3 q_4	s_1 s_2 s_3 s_4
1 2	0	8 10 8 4		
1 3	0	8 10 8 4	4 8 8 10	
1 4	0	8 10 8 4	4 10	2 1
1 5	2			2 1
2 2	2	3 0 4 0		2 1
2 3	2	3 0 4 0	0 0 3 4	2 1
2 4	2	3 0 4 0	3 4	3 2 4 1

Example: A node m, wishes to share one of its private data item with four other nodes n_1, n_2, n_3 and n_4. So the four nodes are in search of AIDA, S is set to the value 10 (for simplicity) and random numbers selected are 8, 10, 8 and 4 in the first round. n_1 and n_3 chooses 3 and 4 respectively in the second round. n_2 and n_4 in the

second round chooses 0 as they are already assigned in the first round. Each step of the AIDA algorithm is clearly shown in Table III. Finally, the AIDA got is $s_1 = 3$ for n_1, $s_2 = 2$ for n_2, $s_3 = 4$ for node n_3, and $s_4 = 1$ for node n_4.

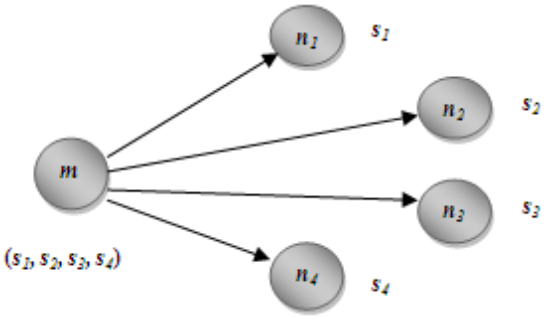

Fig. 1. Data Sharing using AIDA.

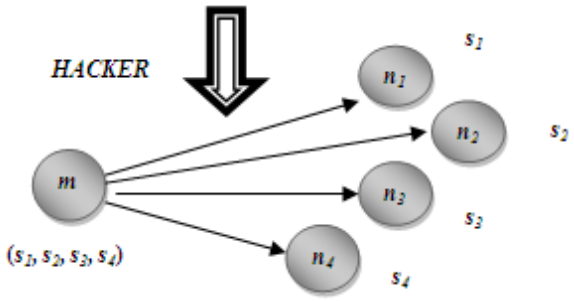

Fig. 2. Hacker attacked AIDA in Data Sharing.

Node m (Sender) is the only node that is aware about all the four IDs. Each of the four nodes is aware only about their own assigned AIDAs. Now, this AIDA got is used as the key for AES encryption and decryption [9]. The private data and destination address undergoes AES encryption and this is done by node m. Since each of the receivers is unaware of others IDs, they are restricted to open their own messages only. Thus the privacy of data and the participants in the group, both the sender and receivers is obtained. The above mentioned system is shown in detail in Fig. 1.

Parameter Analysis (Security): The Detailed AIDA algorithm in my thesis proved that it posses less security.

3.3 New Algorithm

The problem of the AIDA algorithm is solved by the new algorithm. AIDA is a perfect system of data sharing in case of only two domains: privacy preservation and anonymity in ID assignment. AIDA posses less security. The system can be attacked by hackers. The hacker attacked AIDA system is shown in Fig. 2. Security is the most important factor for a system in internet. The new algorithm posses higher security. Security parameter of the new important factor for a system in internet. The new algorithm posses higher security. Security parameter of the new system is provided using SHCS and MAC address verifications. In my thesis, new algorithm uses client server communication with socket.

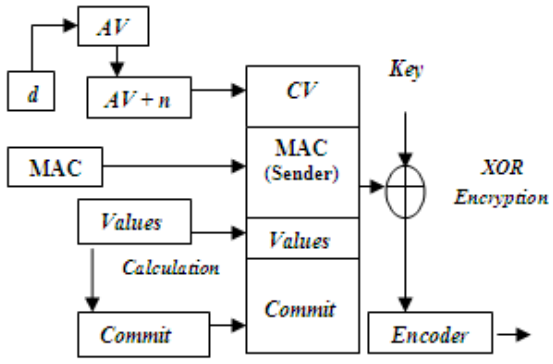

Fig. 3. Private data sharing in new algorithm.

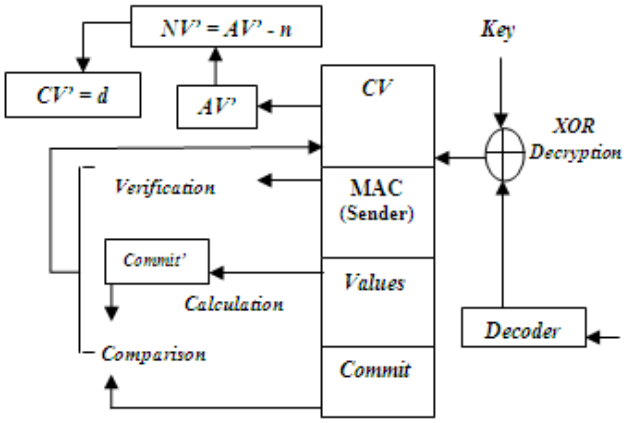

Fig. 4. Private data reception in new algorithm.

Sender Side Algorithm: Say a sender node X, wish to share one of its private data d, with a receiver node Y.

1) Initially, MAC address verification of sender node is done.
2) If step (1) is success, then set a commit value, Commit.
3) Next, each character in the data d, to be shared is converted to its ASCII value (AV).
4) Calculate a new value, NV = AV + n, where n can be any natural number.

5) CV-character version of NV, MAC, and Commit value are encrypted using XOR encryption.
6) The encrypted data is then encoded.
7) Finally, the data is passed to its destination using socket communication.

Firstly, the MAC address of the sender node is verified. This verification is done with the help of MAC address stored by the admin in the database at the time of authorized officer approval. The node is allowed to share its private data only if the above step is a success.

Secondly, the node sets its commit value. The commit value is the final result of a calculation in my thesis. A set of integer values is assigned for this purpose. Now each character of the private data d, is converted to its ASCII code and then it is added with n. n can be of any natural number. Here n = 2 is used. Now the character version of NV is created and called as CV. The new character version of data CV, MAC address of the sender, integer values used for commit generation and the commit value got are encrypted using XOR encryption [10]. The final result is encoded using Base64 Encoder [11], and send to the proper destination using socket communication. Above mentioned algorithm is clearly shown in Fig. 3.

Receiver Side Algorithm: This algorithm explains about the different steps performed at the receiver side when the private data d, shared by X reaches Y.

1) Initially, MAC address verification of receiver node is done.
2) If step (1) is success, then the data received is decoded.
3) Then the output from decoder is passed for the XOR decryption process.
4) Now, calculate the new commit value from the integer values.
5) Check if the new commit value got is equal to the commit set at the sender side.
6) MAC address verification of sender is done
7) If step (5) and (6) are success, only then each character in the data CV, is converted to its ASCII value (AV').
8) Find a new value, NV' = AV' – n, where n can be any natural number.
9) CV'-Character version of NV' is found.
10) CV' = d, this is the private data shared by X.

Firstly, the MAC address of the receiver node is verified. This verification is also done with the help of MAC address stored by the admin in the database at the time of authorized officer approval. The node is only allowed to decode the data if the above step is a success.

Base64 decoder is used for decoding purpose [11]. The result got is then decrypted using XOR decryption [10]. Now the integer values are taken and a commit value is recalculated by the receiver, Commit'. Comparison of Commit and Commit' is done next. Both should be same. Then the MAC address of sender is verified. If both the above mentioned checking is true, only then each character of CV is converted to its ASCII code AV', and n is subtracted from it. n can be of any natural number. Here n = 2 is used. The character version CV' of the new value NV' is established. CV' is the private data d shared by the sender node X. Above mentioned algorithm is clearly shown in Fig. 4.Thus the complete description of the new algorithm is done.

IV. Parameter Analysis

From the very first existing system, privacy preservation is the parameter discussed in detail. Discussions on security of the private data to be shared were not up to the level. AIDA is the latest scheme in the literature. In my thesis, both AIDA and the new algorithm are implemented. AIDA posses the privacy preservation parameter but it lacked the level of security. New algorithm is build on top of MAC address verification and SHCS verification.

Table IV Security Analysis

Algorithm	Security Implementing Processes
AIDA Algorithm	1. AIDA key used for AES Encryption & Decryption
New Algorithm	1. MAC Address Verification 2. SHCS 3. XOR Encryption & Decryption 4. Base64 Encoding & Decoding

A sender node is allowed to share its private data only after the MAC address verification. Here the MAC address of the sender node itself is checked. This improves the security of authorized sender nodes. After this verification a commit value is calculated from a set of integer values. All these data are encrypted along with the private data. At the receiver side, MAC address verification of receiver node is done. The node is allowed to decode and decrypt the data only if the MAC address verification is a success. This improves the security of authorized receiver nodes. Now after the decryption, the SHCS verification and MAC address verification of sender is done. The private data shared can only be viewed if the above mentioned verification

schemes are success. The MAC address verifications at the two sides also preserve the privacy parameter in the new algorithm. Above mentioned analysis is shown in Table IV.

Thus it is completely clear that the new algorithm has got an enhanced security parameter and there by prevents the hackers. New algorithm posses both privacy preservation and data sharing security.

V. Conclusion and Future Work

My thesis addressed the problem of security in private data sharing. The latest scheme in literature AIDA, posses privacy preservation but less security during data sharing. A new algorithm is developed for the easy and secure sharing of unknown private data. New algorithm combines cryptographic primitives such as Commitment scheme (SHCS), XOR encryption and decryption, and Base64 encoder and decoder. My thesis examined both AIDA and the new algorithm, and arrived at the conclusion that the new algorithm has improved the data sharing security with MAC address and SHCS verifications. Thus the hackers are prevented from viewing the shared data in the private data sharing process. Prevention of attackers is the future work for my thesis.

References

[1] A. Friedman, R. Wolff, and A. Schuster, "Providing k anonymity in data mining," VLDB Journal, vol. 17, no. 4, pp. 789–804, Jul. 2008.
[2] F. Baiardi, A. Falleni, R. Granchi, F. Martinelli, M. Petrocchi, and A. Vaccarelli, "Seas, a secure e-voting protocol: Design and implementation," Comput. Security, vol. 24, no. 8, pp. 642–652, Nov. 2005.
[3] D. Chaum, "Untraceable electronic mail, return address and digital pseudonyms," Commun. ACM, vol. 24, no. 2, pp. 84–88, Feb. 1981.
[4] Q. Xie and U. Hengartner, "Privacy-preserving matchmaking for mobile social networking secure against malicious users," in Proc. 9th Ann. IEEE Conf. Privacy, Security and Trust, Jul. 2011, pp. 252–259.
[5] Larry A. Dunning, "Privacy preserving data sharing with anonymous ID assignment," IEEE Trans. Information Forensics and Security, vol. 8, no. 2, February 2013.
[6] Alejandro Proano and Loukas Lazos, "Packet-Hiding Methods for Preventing Selective Jamming Attacks," IEEE Trans. Dependable and Secure Computing, vol. 9, no. 1, 2012.
[7] A. Shamir, "How to share a secret," Commun. ACM, vol. 22, no. 11, pp. 612–613, 1979.
[8] Medium Access Control Address (MAC Address) – Available: http://en.wikipedia.org/wiki/MAC_address.
[9] William Stallings, "Cryptography and Network Security," 4th edition, Pearson Education.
[10] XOR Encryption and Decryption – Available: http://en.wikipedia.org/wiki/XOR_cipher.
[11] Base64 Encoder and Decoder – Available: http://en.wikipedia.org/wiki/Base64.