IIETA International Information and Engineering Technology Association
Advancing the World of Information and Engineering

Ingénierie des Systèmes d'Information
Vol. 24, No. 5, October, 2019, pp. 481-490
Journal homepage: http://iieta.org/journals/isi

GBMS: A New Centralized Graph Based Mirror System Approach to Prevent Evaders for Data Handling with Arithmetic Coding in Wireless Sensor Networks

Subramanian Balaji[1], Yesudhas Harold Robinson[2*], Enoch Golden Julie[3]

[1] Department of Computer Science and Engineering, Francis Xavier Engineering College, Tirunelveli 627003, India
[2] School of Information Technology and Engineering, Vellore Institute of Technology, Vellore 623014, India
[3] Department of Computer Science and Engineering, Anna University Regional Campus, Tirunelveli 627007, India

Corresponding Author Email: haroldrobinson.y@vit.ac.in

https://doi.org/10.18280/isi.240504

Received: 12 April 2019
Accepted: 9 August 2019

Keywords:
crypto signature, hash function, Skolemization, code conversion, efficiency, security

ABSTRACT

In wireless sensor networks (WSNs), the data packets are aggregated and compressed before being sent to the base station, aiming to keep the data secure in the transmission process. This paper presents an arithmetic coding scheme with graph-based mirror system (GBMS) to guard against intruders. Considering its good encryption effect, the hash function, a.k.a. the one-way function, was adopted to encrypt the original messages. The crypto signature was selected to validate the source and destination with temper-proof hash function. In addition, each message was transformed into a graph through inverse mirror mapping, and the graph was taken as final cipher along with the hash value. All the messages were authenticated and authorized by session id created at the time of user entry. Simulation results show that the proposed scheme achieved an efficiency up to 99% without any loss or addition of information. The research findings protect the message integrity and facilitates data aggregation in the WSNs.

1. INTRODUCTION

In WSNs, the sensor nodes are deployed randomly in huge amount of network for monitoring the environmental information and delivered the data to the base station with the base station via wireless communication. The transmission range is used to transmit the data packet in the network. There are several types of applications are implementing the WSN through wireless communication [1].

In a multi-hop WSN, data attribution of every data packet needs the information about the transmitted data packets to the base station [2]. The information about the transmitted data to the base station utilized the current situation of the initial information that the operation has been implemented through the generating networks [3]. Hence the data attribution needs data trustworthiness. The total size of the data packet doesn't exceed the limit of the assumed size for the individual data packet itself [4]. There will be the storage and resource constraints need the capability to implement the data attribution whenever the total size is huge. Normally, sensor nodes need the energy to transmit the data packets. Energy utilization is also played a vital role to implement the transmission of data packets in the network [5].

For the large-scale wireless sensor network, the data attribution normally could not be directly communicated through the network because of the energy and bandwidth parameters [6]. The encoding techniques are utilized to provide the security based data aggregation and attribution of communicating the data packets in the network [7]. Because of using the light weight methods, the packet drop percentage could be very high [8]. The network topology is also the main constraint to implement the data transmission in the network [9]. For the secured data path, the dictionary related methodology is used to compress the data and reduce the network overload issues [10]. The recent research focuses on the data attribution in the workflow, databases and failed to address the security problems in efficiency. To overcome these problems and to provide the efficiency for data packet transmission, we provide Graph Based Mirror System (GBMS) for reduce evaders in data handling for WSNs. The encryption and decryption techniques are implemented to provide the security and data attribution in the base station.

The specific contribution of the paper is:

i. A Graph Based Mirror System based arithmetic coding methodology for Wireless Sensor Networks to provide the data aggregation.
ii. The base station is utilized the efficient method of decoding and verification of data packets.
iii. A secured method for sharing the shared key within the specific node using arithmetic coding to provide the confidentiality and used to evade the malicious node attacks.
iv. A packet sequence number is generated in a secured way to assist the integrity and data attribution.
v. The performance for the proposed methodology is evaluated with the simulation and valid experiments.

The remainder of this paper is organized as follows: Section 2 presents the detailed study about the related work, Section 3 implements the proposed methodology with valid mathematical contribution and algorithms, Section 4 presents the detailed performance evaluation with performance metrics and finally the paper concludes with proper explanation and future directions.

2. RELATED WORKS

Arithmetic coding is the concept of modifying the original ciphers into numerically encoded with adaptive structure [11].

The individual original symbols could not necessarily translate to a similar code indexed, it is totally encoded. The original input character is referenced by the real number of intervals between 0 and 1. The difference of two values is defined by the specified interval. Here, the High and Low values are initialized by 0 and 1. Each new original symbol is subdivided by the some of the specified intervals. The original symbols are distinct by the probable symbol. The value of precision represents the interval sequence.

The Interval Splitting is fully modified by the Arithmetic Coding is mainly used the Key based Interval Splitting Technique. The periods in Arithmetic Coding are split the cipher span that has been increased to the Arithmetic Coding. It is fully bounded with N-original symbol sequence [12, 13]. The Arithmetic coding symbols are executed 0.5 bits. The encryption level of the function key technique and encoded sequence are split by the level of specific symbols. The Binary static methods can be applied to Arithmetic Coding. The SAC-The Secure Arithmetic Coding, associated intervals are split with a key. The original symbol sequence is permuted, the input binary sequence has been converted as the output binary sequence [14]. The Compression and Security approaches have been executed by the individual systems. The Thwarts Attacks are the concept of to stop something from happening which have been approached the data based on the participation or production transformation or the split period key [15]. For prearranged technique representation, the interval of brace is split and individual split may execute in similar. The identical throughputs should be achieved the arithmetic code conversion [16]. The permutation complexity has been divided into the negligible value [17]. SAC-The Secure Arithmetic Coding conversion with the Adaptive Chosen Cipher-text Attacks, the key vectors can recover the permutation step in the model of complexity O(N) where the symbol N is the progression extent [18]. SAC specifies that is not an application where the hackers or attackers can encompass admittance to the decoded message. Moreover, we have engaged an enhanced representation of the SAC. This situation can protect the adaptive chosen cipher-text attack [19] and also which can be incorporated with the Context-based Cryptosystem [20].

The Modification of the Arithmetic Coding have been advanced the improvement conventional Arithmetic Coding [21, 22]. The Arithmetic Code are randomized Arithmetic Coding (RAC) and the arithmetic code with key-based period splitting are proposed for the 2 methodologies, While the similar key is generated to perform the encryption in the cryptosystem based on the dissimilar type of messages [23] and Adaptive text compression techniques [24]. The security of Arithmetic Code is based on the encryption of the one plain text. By this concept, we show that Randomized Arithmetic Code is not secure even if random key is used to the compression of every single messages [25]. This method describes the first compression and then the encryption, where the cryptography methodology is created by the bitwise-XOR of the compression-based productivity.

Security can be done in Ad hoc network by creating Plundering Track [26], using encryption schemes for user-data [27], Enhanced Cluster based key management Technique [28], Secured cluster formation [29], compression technique [30], establishing pairwise keys with the usage of pre-distribution techniques [31], fuzzy classification is used for memory aided broadcast methods [32].

Energy-aware multipath routing [33], Secure Dominating set construction algorithm [34], Randomized algorithms [35], Tree based data fusion algorithm [36], key scheduling algorithm [37], Tree-Based Opportunistic routing [38], Game theoretic approach [39], Un Observable Secure Proactive routing protocol [40].

In our proposed system, the Arithmetic coding has been encrypted by the randomized matrix, the session key of the text is generated by the system for every translate messages. It is mainly access to symbols conversion in the matrix before the process of encryption technique. The integration of Randomized Matrix and the Code Arithmetic has to be achieved the maximum security with minimum complexity using Channel Aware Routing [41].

3. PROPOSED SYSTEM

3.1 System architecture

In Key based Secure Arithmetic Code can be split into two disjoint sub intervals. Figure 1 demonstrates the operation for the first disjoint interval k_1. Figure 2 demonstrates the operation for the second disjoint interval k_2.

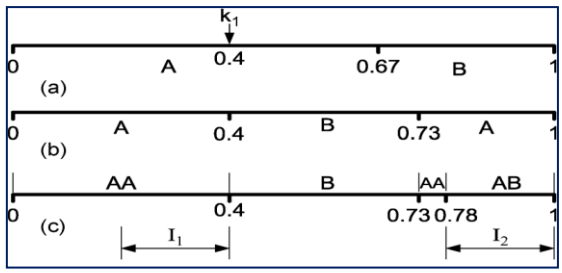

Figure 1. Operation of KSAC for k_1

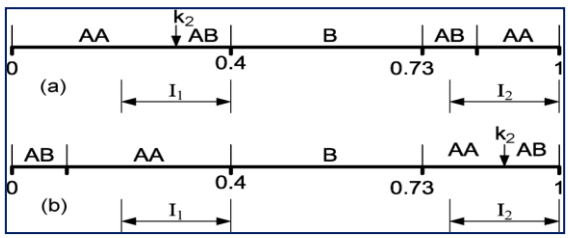

Figure 2. Operations of KSAC for k2

The projected Key Based Secure Arithmetic Code repeatedly uses the given steps.

Step 1: Segmentation of the interval to Arithmetic Code conversion. Exclusive Interval could be executed the union of two different disjoint subintervals. The original symbols are received; who is determined the partition value.

Step 2: The Map key conversions are k € [0, 1] to either interval I_1 or Interval I_2.

Step 3: Implement the splitting of the interval.

Step 4: The original input Symbols have been read, the resulting value of interval is integrated

Step 5: The output will be the different disjoint sub intervals or the contiguous interval.

Figure 3 demonstrates the architecture design for the proposed system GBMS. The attribute set is formulated using the group of related attributes for the security. The attribute set consists of the data. The Central authority and the access policy are the 2 parameters that can be joined using the XOR

operation. The formed access polity is then encrypted using the original method by using the public key and the XOR operation. Using the KSAC model, the cipher text is created using the attribute set, central authority and the access policy.

The final encrypted message is created using the information based on the client and the server with the cipher table. Figure 4 illustrates the model of GBMS with Client server technology.

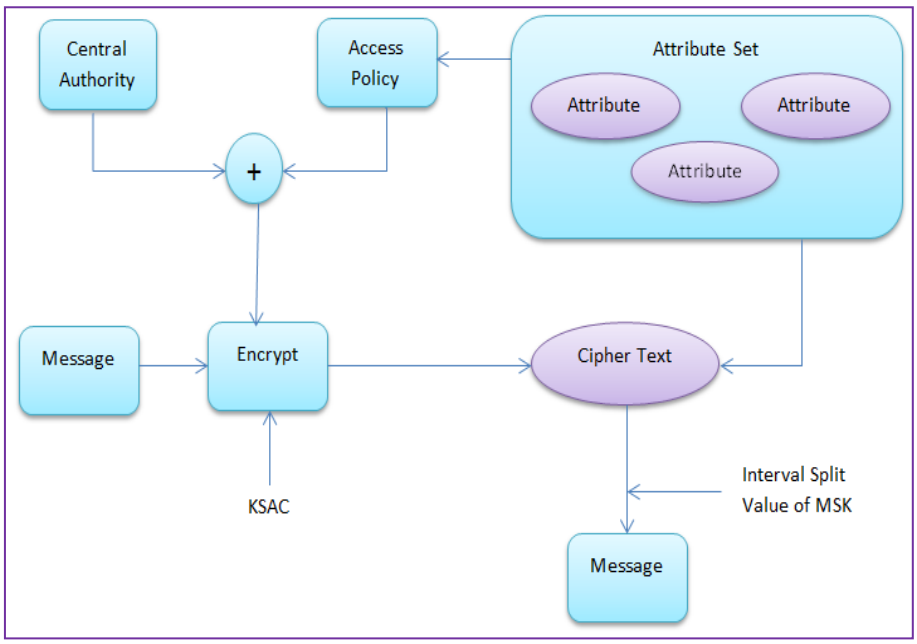

Figure 3. Architecture of GBMS

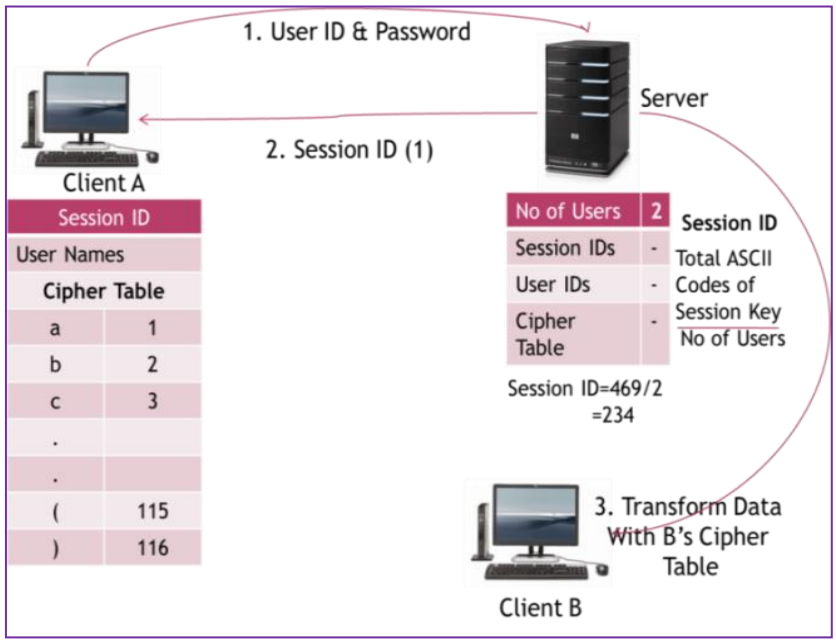

Figure 4. Modeling of GBMS

The client is with the usual attributes like username, password and other details and the one exception is cipher table which is dynamically updated in regular intervals by the consumer development supported on the received information which is session id from the server. The server holds the status of amount of users active in the system and the current session and user id with their corresponding cipher table. The session id is calculated as the ratio of the total ASCII codes of the session key to the no of users.The system is centralized that means all the data is transmitted through the server with the receiver destined format.

3.2 Encryption-GBMS

The message is constructed as usual way which is split into multiples of 3. In case of any deficiency in last three times, the dummy symbols will be added. Derive the ASCII codes for the every character in the message and split each character in the thrice into x and y coordinates. A single thrice forms two joint lines that have to be mirrored using mirror mapping and make the inverse through the inverse mapping. The final transform will be appended with the hash value and transformed into GBMS cipher using cipher table. Figure 5 illustrates the encryption model of GBMS.

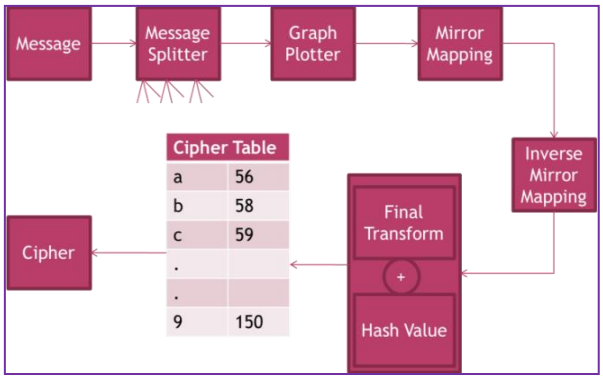

Figure 5. Encryption of GBMS

3.3 Graph plotting and mirror mapping

The single message is split into 3 character block and converted into ASCII codes to form x-y coordinates and mapped onto the graph. The point that joints two characters will take it as center point and it is mirrored based on that point by identifying the difference between the x-y coordinates. The output of this process will produce the intermediate cipher that have to be supplied the input of inverse mirror mapping. Figure 6 demonstrates the Plotting point in the graph based GBMS model which consists of the group of data for graph plotting. The plotted points are then marked in the graph using this methodology. Figure 7 demonstrates the Mirror mapping in the graph based GBMS model.

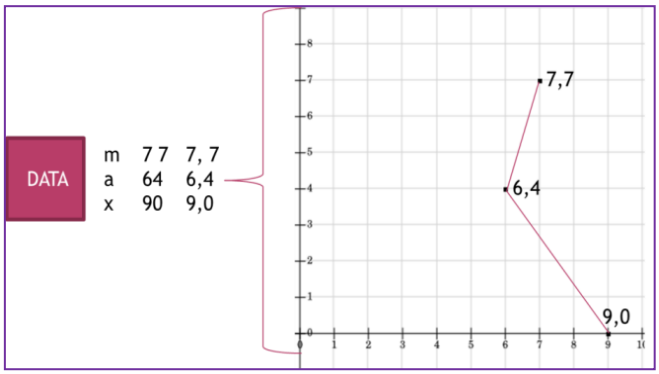

Figure 6. Plotting points in graph

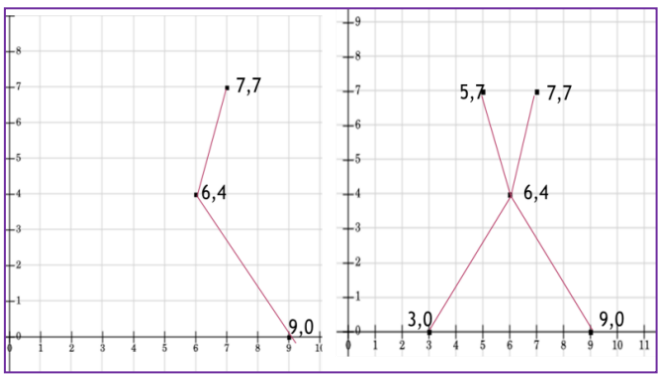

Figure 7. Mirror mapping

3.4 Inverse mirror mapping

Take the topmost point of the mirror image and make it as base point. Use that point to deduce the x-y coordinates for the remaining two characters by finding the difference between xy values. The output of this process will produce the intermediate cipher that have to be fed into as input of final transformation based on cipher table with the hash value. Figure 8 demonstrates the Inverse Mirror mapping.

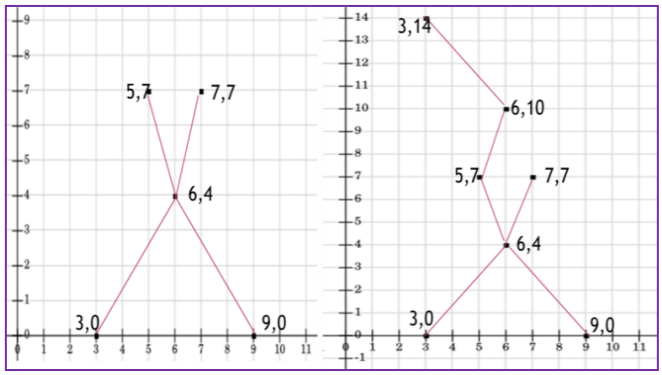

Figure 8. Inverse mirror mapping

3.5 Decryption-GBMS

The received cipher is converted into transformed value based on intended cipher table format created and updated at the regular interval. This value is separated into transformed value and hash value. Then transformed value fed into data splitter which split into three times blocks and maps the inverse image. Then the inverse image is translated into mirrored image and then back to original image by identifying the base points x and y. Implement the reverse process with the ASCII process and calculate the hash value with original received value. Figure 9 demonstrates the Decryption model for GBMS.

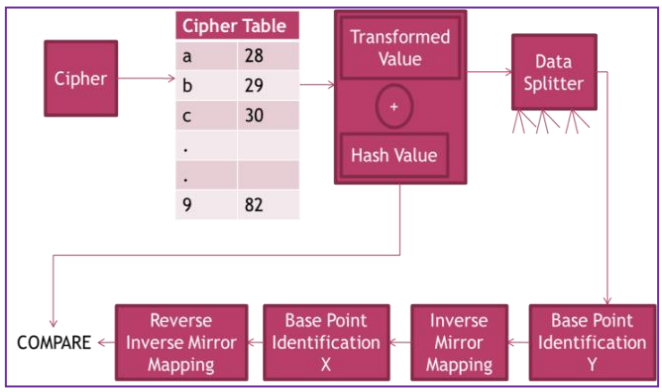

Figure 9. Decryption-GBMS

3.6 Algorithm-GBMS

3.6.1 Encryption

Input: Message
Encrypt(Message)
begin
Split(Message)
Interval []=ASCII(Message) //multiples of 3
split Interval into Interval1 & Interval2
for i 1 to 3 do
begin
GPlot=graph[Int1 as x,Int2 as y]
find short x-Base X
mirror(x,y)

find long y-Base Y
Inversemir(x,y)
end
Lookup_Cipher_Table (x, y)
Append (Hash Value)
Return Cipher
end
Input: ID->Session ID
Begin
For i=0 to 128
Cipher_Table[i]=ID
ID++
end
Input: Username & Password
AutoGenerateKey(username+password length)
SessionID=Tot.ASCII values session key%No of Users
End

3.6.2 Hash function

Function f(Key, K)
Initialization variable a,b,c
(a,b,c) = Hash (k + i)
i = i + 1
if z < f(a,b)
$B(a_{Hash}, b_{Hash})$ = { Arithmetic Coding of id_1, $id_2(a,b)$: |(a_{Hash}, b_{Hash}), (a,b)|}

3.6.3 Regular queue

Message m, Group Member M_1, Random Arithmetic Variable a,b,c,d and Integer values x,y,z.

Manipulate {a,b,c,d} using Key_K and function f
$a = RMAC^{\alpha} \bmod p_k$
$b = (RMAC \times X) \bmod p_k$
$c = (f - y) \bmod p_k$
$d = (g_{id}^{x-y}) \bmod p_k$
$Hash(m) = RMAC \times x_i + z \times S_{id} \bmod p_k$
Group signature for message m_1
{h(m), h(m1),a,b,c,d}
Step 1: Calculate $\alpha = E^C \times RMAC_{id1} \times D \bmod p_k$
Step 2: Calculate $H_i = \alpha_i \times a \bmod p_k$
Step 3: Check the Group Signature Relation $\alpha^{hash(m)} \equiv H_i^{RMAC} \times R^{Sid} \bmod p_k$

3.6.4 Validation

$\alpha^{hash(m)} \equiv \alpha^{(RMAC \times xi + Sid)}$
$\alpha^{hash(m)} \equiv (Hash_i^{RMAC} \times RMAC^{Sid}) \bmod p_k$

3.6.5 Procedure for anomaly detection

Step 1: Create group id, RMAC id, Modulus of Private key K_1
Step 2: $group_{id}^{k1} \times group_{id}^{-K} \times group_{id}^{xi-ki} \bmod p_k$
Step 3: groupid (xi + sid + (RMAC id1 x SIV) $\bmod p_k$
Step 4: $(group_{id}^{Sid} \times p_k^{id1})^{xi} \bmod p_k$

3.6.6 Digital Signature for Solemnizations

Each Encrypted key $f(x_i)$ and randomized private key k_i, $1< K_i<n-1$, to manipulate attacker identifier (V_i, $Sign_i$) for m.

Result $Interval_i$ = ($xRInterval_i$, $yRInterval_i$) K_i x Goal Based Interval

RMAC = xResult $Interval_i$ mod n and is deminstrated in Eq. (1).

$$RMAC = Key_i + f(x_i)\left[\prod_{Sid=1}^{ti} \frac{-x_j}{x_i - x_j}\right] \quad (1)$$

The variable y_k, n_k and calculate
$RMAC_A \leftarrow E\ (Hash(y_k \times ID_A)^P + P\ pub, T_A)$
$y_k^{hash\ k} V[\ ] \leftarrow Hash(RMAC_A, y_k, ID_A)$

Message received from the original source and the input id is generated to the cipher. Received id is created to the cipher table and mapped the numbers until the end of symbol is reached. Finally, the ASCII form of conversion can be entered as message. After the data split, use the ASCII codes to divide the numbers as x and y cords for graph plotting. Then map the image in the mirror form and then transformed into inverse mirror form. Transform the last form using cipher table and append the hash value. Figure 10 demonstrates the Algorithm description of GBMS. Figure 11 demonstrates the architecture design and Figure 12 illustrates the Matching Data type and Size Deduction.

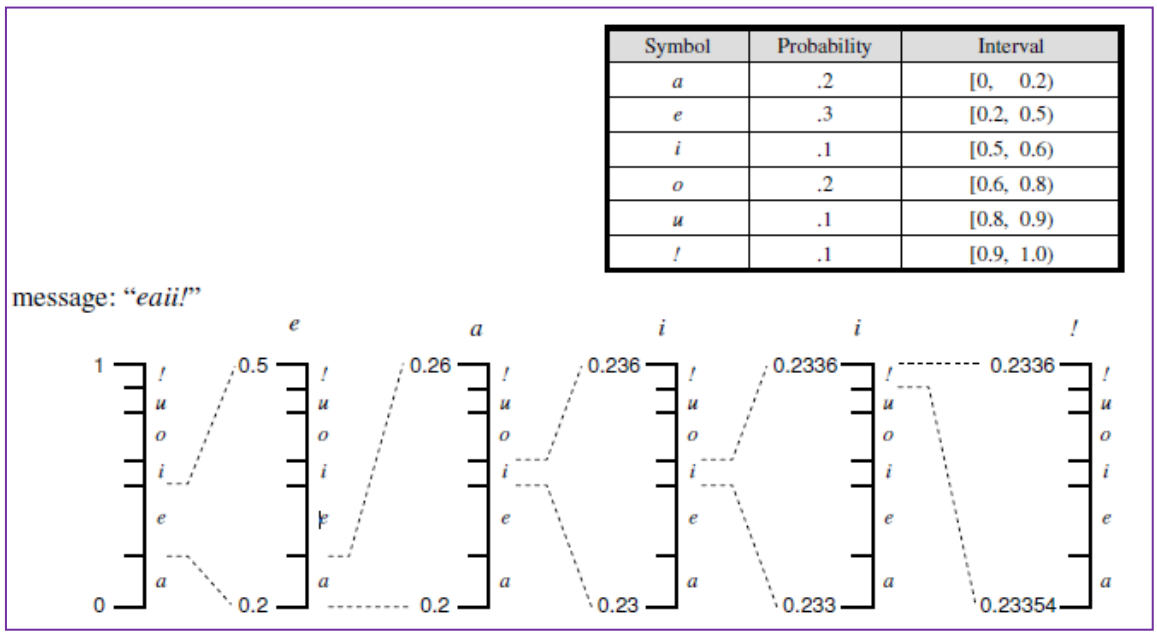

Figure 10. GBMS algorithm description

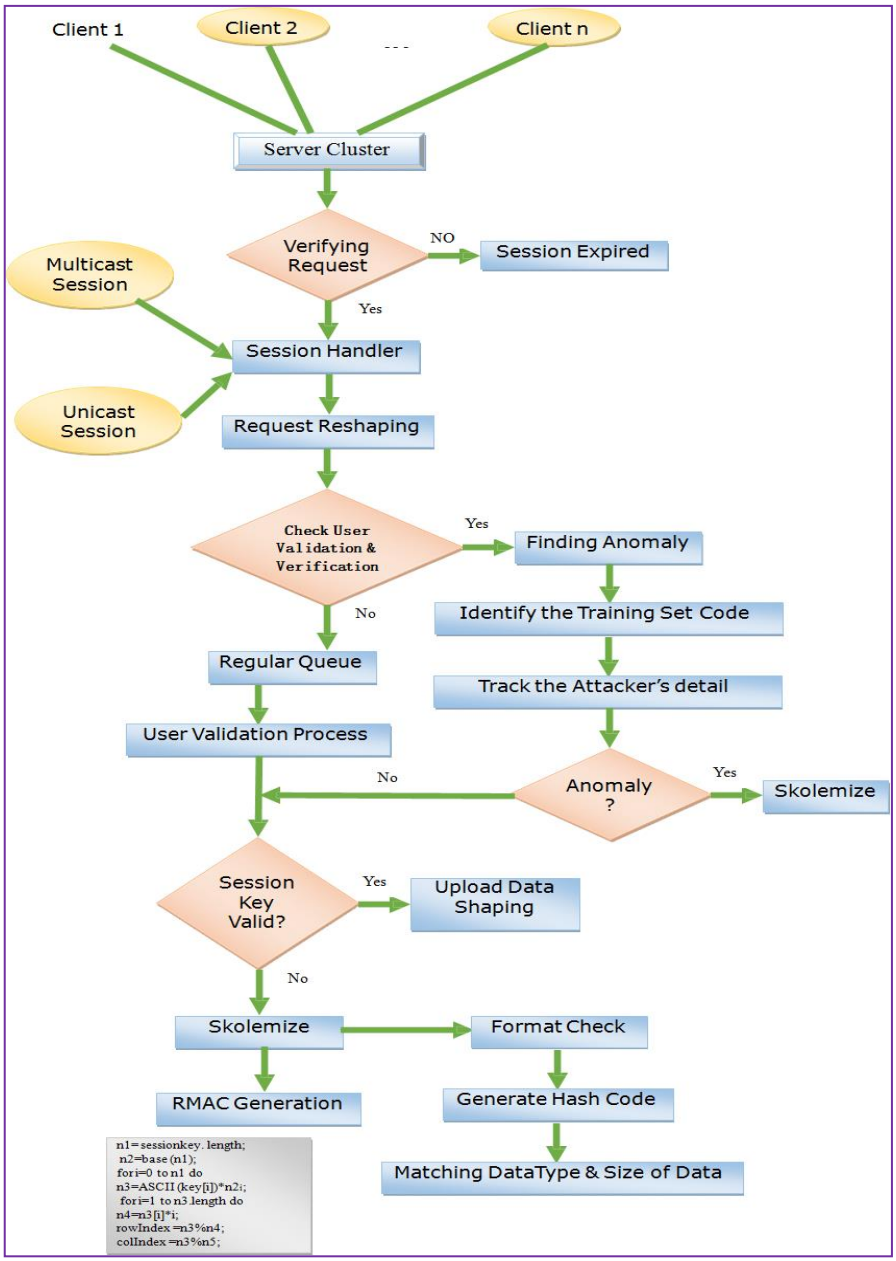

Figure 11. Architectural design

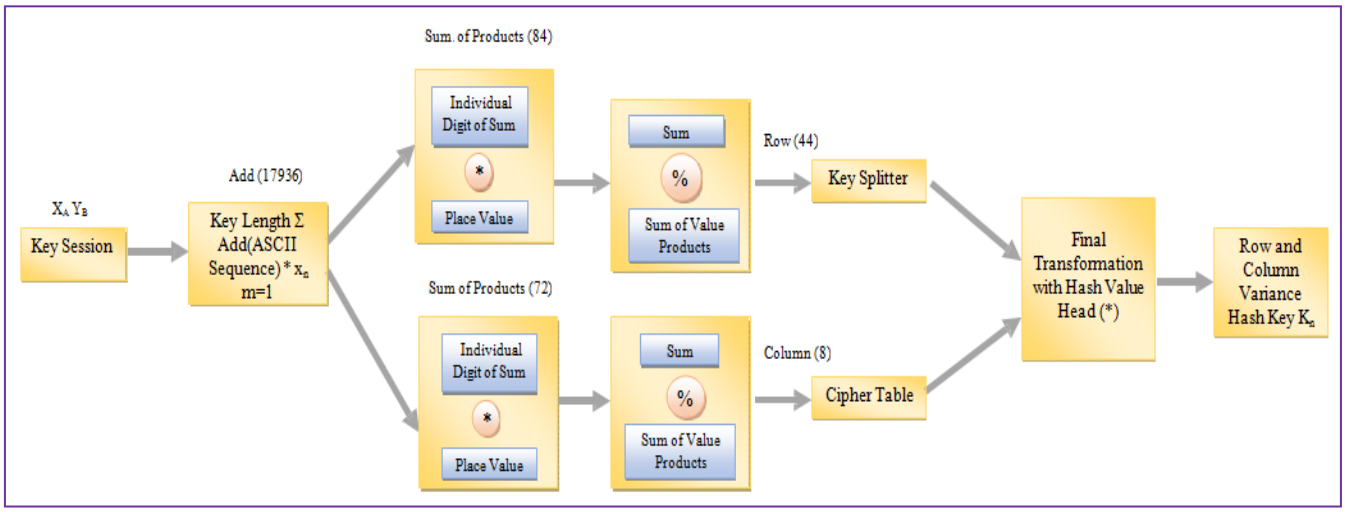

Figure 12. Matching data type and size deduction

$$T_x x\,(a_i) = \sum_{k=1}^{i-1} P(x = k) + \frac{1}{2} P(x = i) \qquad (2)$$

The interval for the midpoint is used as the Tag T with Function which will be used in GBMS arithmetic coding interval and is illustrated in Eq. (2). For Graphical ordering, for all the messages of m, the Tag function will be as follows in Eq. (3), Eq. (4), Eq. (5).

$$T_x\,(a_i) = \sum_{y< xi}^{i-1} P(y) + \frac{1}{2} P(x_i) \qquad (3)$$

$$len^{(k)} = len^{(k-1)} + (union^{(k-1)} - len^{(k-1)})F_X(x_k - 1) \qquad (4)$$

$$7nion^{(k)} = len^{(k-1)} + (union^{(k-1)} - len^{(k-1)})F_X(x_k) \qquad (5)$$

If mid-point is used in the equation, the Tag function is used to find the mid value of ai will be computed in Eq. (6)

$$T_x\,(a_i) = \frac{(len^{(k)} + union^{(k)})}{2} \qquad (6)$$

3.6.7 Crypto signature

To Encrypt Message $\{1,0\}^k$

Public Key and valid identity key and sender's private key

RMAC ← $y_B{}^h$

Cipher ← Hash (RMAC) ⊕ message

Hash ← $Hash_1$ (RMAC, Message, y_A, ID_A, y_B, Sid_A)

Arithmetic Code ← θ / ($Hash_1$ + x_A)

Crypto Signature is the triple (Cipher, Hash, Arithmetic Code)

RMAC ← $y_A{}^{(xB\ x\ z)}$ x $y_B{}^{(Hash\ x\ z)}$

Message ← $Hash_2$(RMAC) ⊕ Cipher

v ← $Hash_3$(RMAC, Message, y_A, ID_A, y_B, Sid_A)

The Receiver can accept the message from the sender if and only if v = hash

Public Key (P_k) & Private key (Pr_k)

1. Logical Validation ← $y_k{}^{xv\ Rv}$, L_{RMAC} ← $y_v{}^{\theta/2}$
2. Session Validation ← $hash_1(l_v, y_v, ID_v, y_R, ID_{RMAC})$
3. Certificate Encryption ← $$\frac{x_{vav}(hash_1(y_v, ID_v)P + P_{pub})}{t_v(x_v - x_v hash_v)}$$
4. Certificate Decryption ← $$\frac{x_{RMAC}\, a_{RMAC}(hash_1(y_{RMAC}, ID_{RMAC})P + P_{pub})}{t_{RMAC}(u_{RMAC} - x_{RMAC} hash_{RMAC})}$$

4. PERFORMANCE EVALUATION

The performance Evaluation has been performed using Network Simulator 2. The parameter metrics for analyzing the proposed work are efficiency, packet loss, transmission rate, connectivity delay, average code conversion, process path length, number of IP packets sent. The Enhanced Secure Arithmetic Code (ESAC) [42] has been permuted for the convolution O (Nc) + 3 where Nc is the cipher span. We have analyzed the ESAC hack convolution with extended Randomized Arithmetic Code (XRMAC) [43], Randomized Arithmetic Code (RMAC) [44] exclusively accomplishes 49.89%. Finally, we have improved 20% protection in the hacker renovation region of Arithmetic Code than accomplished in the related method (ESAC). Table 1 demonstrates the Comparison between Arithmetic Code and GBMS with the Hacking Complexity.

Table 1. Comparison between arithmetic code and GBMS with the hacking complexity

TEXT LENGTH	CODE WORD		EFFICIENCY GBMS
	AC	GBMS	
100	100	106	94.34
200	200	206	97.09
300	300	306	98.04
400	400	406	98.52
500	500	506	98.81
600	600	606	99.01
700	700	706	99.15
800	800	806	99.26
900	900	906	99.34
1000	1000	1006	99.4
1100	1100	1106	99.46
1200	1200	1206	99.5

The Length of the Text (T) =700

Max. Transformed Text length=T+4=704

No of Control messages=2

Total length of Generated =704

Cipher (GBMS)

Hacking Time (AC) =O (N_c) = 700

Hacking Time (GBMS) = O (N_c) + 2 = 704 + 2 = 706

$$\text{Efficiency} = \frac{\text{Hacking complexity (AC)} * 100}{\text{Hacking complexity (GBMS)}} \qquad (7)$$

$$\text{Efficiency} = \frac{700 * 100}{706} = 99.15\%$$

The Efficiency is computed using Hacking complexity in Eq. (7).

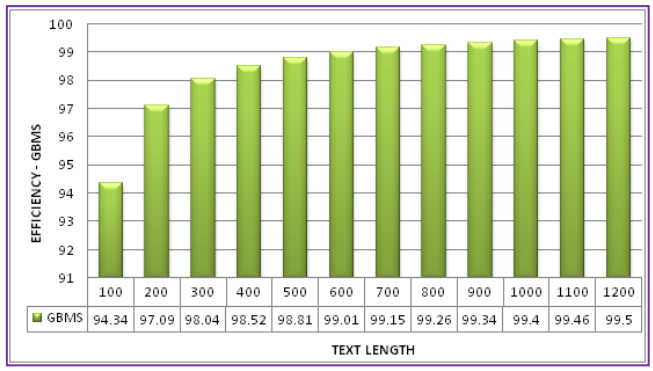

Figure 13. Efficiency between AC & GBMS

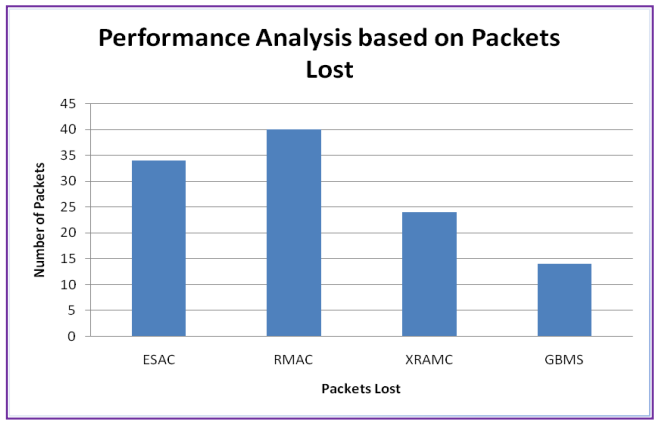

Figure 14. Performance analysis based on packets lost

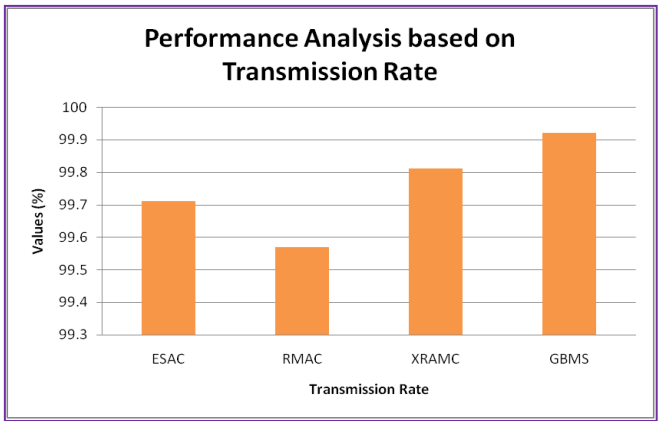

Figure 15. Performance analysis based on transmission rate

The Arithmetic Code Conversion using GBMS is exclusively increased to 99%. Figure 13 demonstrates the performance evaluation of Efficiency between AC & GBMS, Figure 14 illustrates the Performance Analysis based on Packets Lost, Figure 15 shows Performance Analysis based on Transmission Rate, Figure 16 shows Average Code Conversions, Figure 17 illustrates Number of IP Packets Sent, Figure 18 demonstrates Number of Connectivity Delay and Figure 19 demonstrates the Number of Hops vs. Process Path Length.

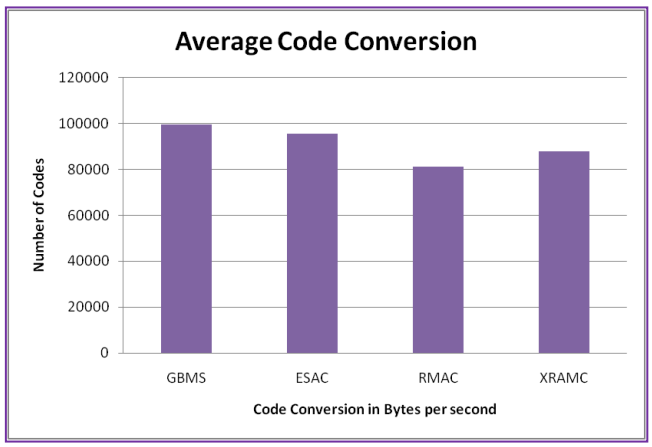

Figure 16. Average code conversions

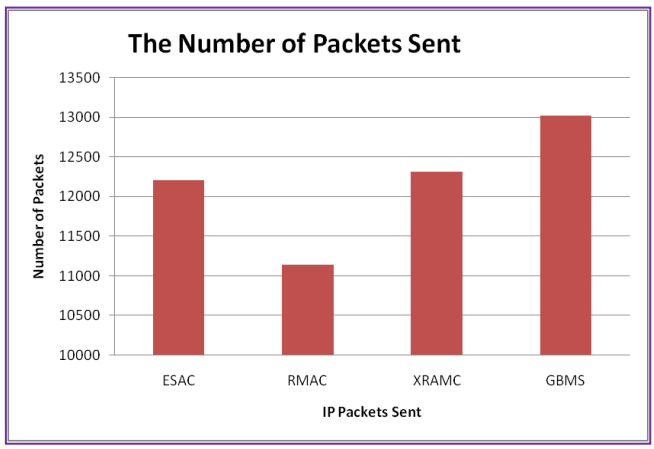

Figure 17. Number of IP packets sent

The performance Evaluation has showed that the GBMS has performed well in various parameters of Transmission Rate, Packets Lost, Average Code Conversions, IP Packets Sent, Connectivity Delay, Number of Hops and Process Path Length.

The performance has been evaluated based on other 3 related schemes such as Randomized Matrix Arithmetic Coding (RMAC), ESAC based Channel Aware Routing with routing handoff, Extended Randomized Matrix Arithmetic Coding (XRMAC).

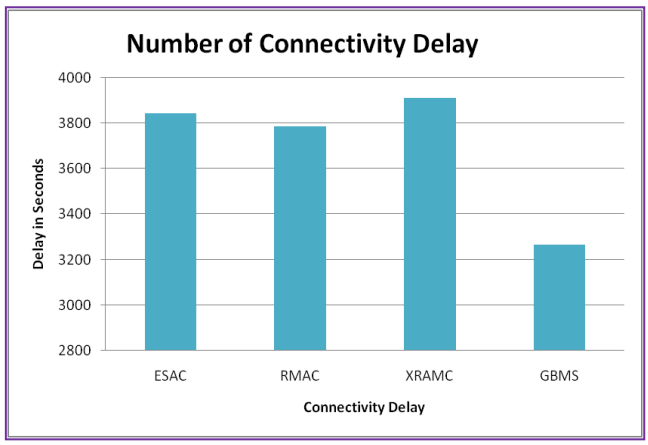

Figure 18. Number of connectivity delay

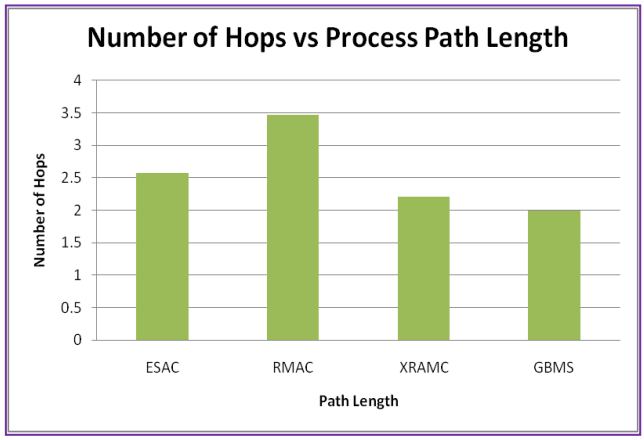

Figure 19. Number of Hops vs Process Path Length

5. CONCLUSIONS

The proposed GBMS system can be used to increase the encryption value in the text and decryption value in the text has been dynamically updated at the regular interval splitting by the server while processing the data from one end to another end in the wireless sensor networks. The WSN demonstrates the aggregation of data and data attribution from the source node to the sink node with providing the quality of service in the network. The data packet uses mirror and inverse transformations methodology to minimize the code complexity instead of using simple arithmetic calculations which increases the security and also improves the efficiency by reducing the length of the cipher-text. In future enhancement, we are going to concentrate additional security and design of the algorithm to transfer the data server in traffic than the previous systems.

The Future direction of this work will implement the security in the wireless systems like Internet of Things, mobile users and real-time systems.

REFERENCES

[1] Wang, C., Hussain, S.R., Bertino, E. (2016). Dictionary Based secure provenance compression for wireless sensor networks. IEEE Transactions on Parallel and Distributed Systems, 27(2): 405-418. http://dx.doi.org/10.1109/TPDS.2015.2402156

[2] Sultana, S., Ghinita, G., Bertino, E., Shehab, M. (2014). A lightweight secure scheme for detecting provenance forgery and packet drop attacks in wireless sensor networks. IEEE Transactions on Dependable and Secure Computing, 12(3): 256-269. http://dx.doi.org/10.1109/TDSC.2013.44

[3] Xu, Y., Wang, J., Wu, Q., Anpalagan, A., Yao, Y.D. (2012). Opportunistic spectrum access in unknown dynamic environment: A game-theoretic stochastic learning solution. IEEE Trans. Wirel. Commun., 11(4): 1380-1391. http://dx.doi.org/10.1109/TWC.2012.020812.110025

[4] Lim, H.S., Moon, Y.S., Bertino, E. (2010). Provenance-based trustworthiness assessment in sensor networks. In Proceedings of the Seventh International Workshop on Data Management for Sensor Networks, pp. 2-7. http://dx.doi.org/10.1145/1858158.1858162

[5] Alam, S.I., Fahmy, S. (2014). A practical approach for provenance transmission in wireless sensor networks. Ad Hoc Networks, 16(1): 28-45. http://dx.doi.org/10.1016/j.adhoc.2013.12.001

[6] Harold Robinson, Y., Balaji, S., Golden Julie, E. (2019). PSOBLAP: particle swarm optimization-based bandwidth and link availability prediction algorithm for multipath routing in Mobile Ad Hoc networks. Wireless Personal Communications, 106(4): 2261-2289. http://dx.doi.org/10.1007/s11277-018-5941-9

[7] Madden, S., Franklin, M.J., Hellerstein, J.M., Hong, W. (2002). TAG: a tiny aggregation service for ad-hoc sensor networks. ACMSIGOPS Operating Systems Review, 36(SI): 131-146. http://dx.doi.org/10.1145/844128.844142

[8] Wang, C., Bertino, E. (2017). Sensor network provenance compression using dynamic bayesian networks. ACM Transactions on Sensor Networks, 13(1): 5. http://dx.doi.org/10.1145/2997653

[9] Hussain, S.R., Wang, C., Sultana, S., Bertino, E. (2014). Secure data provenance compression using arithmetic coding in wireless sensor networks. In Proceedings of the 2014 IEEE International Performance Computing and Communications Conference (IPCCC), Austin, TX, USA, pp. 1–10. http://dx.doi.org/10.1109/PCCC.2014.7017068

[10] Harold Robinson, Y., Golden Julie, E. (2019). MTPKM: Multipart trust based public key management technique to reduce security vulnerability in mobile Ad-Hoc networks, Wireless Personal Communications, 109(2). https://doi.org/10.1007/s11277-019-06588-4

[11] Zhou, Q., Wong, K., Liao, X., Hu, Y. (2011). On the security of multiple Huffman table based encryption. Journal of Visual Communication and Image Representation, 22(1): 85-92. http://dx.doi.org/10.1016/j.jvcir.2010.10.007

[12] Zhu, Z., Zhang, W., Wong, K., Yu, H. (2011). A chaos-based symmetric image encryption scheme using a bit-level permutation. Information Sciences, 181(6): 1171-1186. http://dx.doi.org/10.1016/j.ins.2010.11.009

[13] Zhou, J.T., Au, O.C., Wong, P.H.W. (2009). Adaptive chosen-cipher text attack on secure arithmetic coding. IEEE Trans. Signal Process., 57(5): 1825-1838. http://dx.doi.org/10.1109/TSP.2009.2013901

[14] Katti, R.S., Srinivasan, S.K., Vosoughi, A. (2011). On the security of randomized arithmetic codes against ciphertext-only attacks. Information Forensics and Security, Information Forensics and Security, 6(1): 19-27. http://dx.doi.org/10.1109/TIFS.2010.2096809

[15] Hao, X.C., Gong, Q.Q., Hou, S., Liu, B. (2014). Joint channel allocation and power control optimal algorithm based on non-cooperative game in wireless sensor networks. Wireless Personal Communication, 78(2). http://dx.doi.org/10.1007/s11277-014-1800-5

[16] Miao, X.N., Xu, G. (2013). Cooperative differential game model based on trade-off between energy and delay for wireless sensor networks. Annals of Operations Research- Springer, 206(1). http://dx.doi.org/10.1007/s10479-013-1354-z

[17] Balaji, S., Golden Julie, E., Harold Robinson, Y., Kumar, R., Thong, P.H., Son, L.H. (2019). Design of a security-aware routing scheme in Mobile Ad-hoc Network using Repeated Game Model. Computer Standards & Interfaces, 66: 103358. http://dx.doi.org/10.1016/j.csi.2019.103358

[18] Kim, H., Wen, J.T., Villasenor, J.D. (2007). Secure arithmetic coding. IEEE Transactions on Signal Processing, 55(5): 2263-2272. http://dx.doi.org/10.1109/TSP.2007.892710

[19] Bergen, H.A., Hogan, J.M. (1993). A chosen plaintext attack on an adaptive arithmetic coding compression algorithm. Comput. Secur., 12: 157-167. https://doi.org/10.1016/0167-4048(93)90099-Q

[20] Bergen, H.A., Hogan, J. M. (1992). Data security in a fixed-model arithmetic coding compression algorithm. Comput. Secur., 11: 445-461. http://dx.doi.org/10.1016/0167-4048(92)90011-F

[21] Wen, J.T., Kim, H., Villasenor, J.D. (2006). Binary arithmetic coding with key-based interval splitting. IEEE Signal Process. Lett., 13: 69-72. http://dx.doi.org/10.1109/LSP.2005.861589

[22] Senturk, I.F., Akkaya, K., Yilmaz, S., (2014). Relay placement for restoring connectivity in partitioned wireless sensor networks under limited information. Ad Hoc Networks, 13(Part B): 487-503. http://dx.doi.org/10.1016/j.adhoc.2013.09.005

[23] Safi, Q.G.K., Luo, S., Wei, C., Pan, L., Yan, G. (2018). Cloud-based security and privacy-aware information dissemination over ubiquitous VANETs. Computer Standards & Interfaces, 56: 107-115. http://dx.doi.org/10.1016/j.csi.2017.09.009

[24] Witten, I.H., Clearly, J.G. (1988). On the privacy offered by adaptive text compression. Comput. Secur., 7: 397-408. https://doi.org/10.1016/0167-4048(88)90580-9

[25] Kusyk, J., Cem, S.S., Umit Uyar, M., Urrea, E., Gundry, S. (2011). Self-organization of nodes in mobile ad hoc networks using evolutionary games and genetic algorithm. Journal of Advanced Research, 2(3): 253-264. http://dx.doi.org/10.1016/j.jare.2011.04.006

[26] Balaji, S., Rajaram, M. (2016). SIPTAN: Securing inimitable and plundering track for Ad Hoc network. Wireless Personal Communications, 90(2). http://dx.doi.org/10.1007/s11277-016-3187-y

[27] Balaji, S., Golden Julie, E., Harold Robinson, Y. (2019).

Development of fuzzy based energy efficient cluster routing protocol to increase the lifetime of wireless sensor networks. Mobile Networks & Applications, 24(2): 1-13. https://doi.org/10.1007/s11036-017-0913-y

[28] Bae, S.H., Howe, B. (2015). Gossipmap: A distributed community detection algorithm for billion-edge directed graphs. Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis, Article No. 27. http://doi.acm.org/10.1145/2807591.2807668

[29] Harold Robinson, Y., Balaji, S., Golden Julie, E. (2019). FPSOEE: Fuzzy-enabled particle swarm optimization-based energy-efficient algorithm in mobile ad-hoc networks. Journal of Intelligent & Fuzzy Systems, IOS Press, 36(4): 3541-3553. http://dx.doi.org/10.3233/JIFS-181472

[30] Duan, J., Gao, D., Yang, D., Foh, C.H., Chen, H.H. (2014). An energy-aware trust derivation scheme with game theoretic approach in wireless sensor networks for IoT applications. IEEE Journal of Internet of Things, 1(1): 58-69. http://dx.doi.org/10.1109/JIOT.2014.2314132

[31] Golden Julie, E., Tamil Selvi, S., Harold Robinson, Y. (2016). Performance analysis of energy efficient virtual back bone path based cluster routing protocol for WSN. Wireless Personal Communications, 91(3): 1171-189. http://dx.doi.org/10.1007/s11277-016-3520-5

[32] Robinson Harold, Y., Rajaram, M. (2016). A memory aided broadcast mechanism with fuzzy classification on a device-to-device mobile Ad Hoc network. Wireless Personal Communications, 90(2): 1-23. http://dx.doi.org/10.1007/s11277-016-3213-0

[33] Harold Robinson, Y., Rajaram, M. (2015). Energy-aware multipath routing scheme based on particle swarm optimization in mobile ad hoc networks. The Scientific World Journal, 2015: 9 pages. https://doi.org/10.1155/2015/284276

[34] Ayyasamy, A., Venkatachalapathy, K. (2015). Context aware adaptive fuzzy based QoS routing scheme for streaming services over MANETs. Wireless Networks, 21(2): 421-430. http://dx.doi.org/10.1007/s11276-014-0801-3

[35] Shamshirband, S., Patel, A., Anuar, N.B., Kiah, M.L.M., Abraham, A. (2014). Cooperative game theoretic approach using fuzzy Q-learning for detecting and preventing intrusions in wireless sensor networks. Engineering Application on Artificial Intelligent, 32: 228-241. http://dx.doi.org/10.1016/j.engappai.2014.02.001

[36] Harold Robinson, Y., Balaji, S., Golden Julie, E. (2019). Design of a buffer enabled ad hoc on-demand multipath distance vector routing protocol for improving throughput in mobile Ad hoc networks. Wireless Personal Communications, 106(4): 2053-2078. https://doi.org/10.1007/s11277-018-5925-9

[37] Li, Z., Shen, H. (2012). Game-theoretic analysis of cooperation incentive strategies in mobile Ad Hoc networks. IEEE Transactions on Mobile Computing, 11(8): 78-86. http://dx.doi.org/10.1109/TMC.2011.151

[38] Harold Robinson, Y., Golden Julie, E., Balaji, S., Ayyasamy, A. (2016), Energy aware clustering scheme in wireless sensor network using neuro-fuzzy approach. Wireless Personal Communications, 95(2): 703-721. http://dx.doi.org/10.1007/s11277-016-3793-8

[39] Ahmad, B., Jian, W., Ali, Z.A. Hybrid anomaly detection by using clustering for wireless sensor network. Wireless Personal Communications, 106(4): 1841. https://doi.org/10.1007/s11277-018-5721-6

[40] Jahani, A., Khanli, L.M., Hagh, M.T., Badamchizadeh, M.A. (2019). EE-CTA: Energy efficient, concurrent and topology-aware virtual network embedding as a multi-objective optimization problem. Computer Standards & Interfaces, 66: 10331. https://doi.org/10.1016/j.csi.2019.04.010

[41] Chen, X.Q., Jones, H.M., Jayalath, D. (2011). Channel aware routing in MANETS with route handoff. IEEE Trans. Mobile Computing., 10(1): 108-120. http://dx.doi.org/10.1109/TMC.2010.144

[42] Kavitha, V., Balaji, S. (2011). ESAC based channel aware routing using route handoff. International Journal on Computer Science and Engineering (IJCSE), 3(3): 1260-1269.

[43] Rajaram, M., Balaji, S., Jeeva, R. (2013). XRMAC-an extended RMAC scheme to evade hacking by dynamic sizing. 2013 Fifth International Conference on Advanced Computing (ICoAC), Chennai, India. http://dx.doi.org/10.1109/ICoAC.2013.6921944

[44] Kavitha, V., Balaji, S., Jeeva, R. (2011). RMAC-A new encryption scheme for arithmetic coding to evade CCA attacks. 2011 Third International Conference on Advanced Computing, Chennai, India. http://dx.doi.org/10.1109/ICoAC.2011.6165170

NOMENCLATURE

$f(x_i)$	function value of(x_i)
k_i	randomized private key
V_i	attacker identifier
$Sign_i$	Signature
T	Tag
$T_x\ (a_i)$	mid-point
P_k	Public Key
Pr_k	Private key