INTERNATIONAL JOURNAL OF ADVANCE RESEARCH, IDEAS AND INNOVATIONS IN TECHNOLOGY

ISSN: 2454-132X
Impact factor: 6.078
(Volume 6, Issue 5)
Available online at: www.ijariit.com

Data and key encapsulation for the smart grid using improved AES and difference expansion algorithm

Rajdeep Kaur
manavgill71@gmail.com
Adesh Institute of Engineering and Technology, Faridkot, Punjab

Puneet Jain
puneetjain988@gmail.com
Adesh Institute of Engineering and Technology, Faridkot, Punjab

Navdeep Kaur Jhajj
japjideepk@gmail.com
Shaheed Bhagat Singh State Technical Campus Ferozepur

ABSTRACT

Smart Grid (SG) is an advanced electrical power grid. SG collects the user's smart meter data over the public network. After that, based on the data, generate the bills, and predict the electricity demand. The data contains sensitive information about users. Thus, the privacy of the users is required for better assessment. To preserve the privacy of the user, symmetric cryptography algorithms are used. In the symmetric algorithm, the same key is required for encryption/decryption purposes. Thus, the key is communicated to the receiver with the encrypted data. Thus, it is prone to eavesdropping and tampering attacks. Therefore, in this paper, we have designed data and key encapsulation algorithm that secure the data as well as the key. We have used the Advanced Encryption Standard (AES) algorithm for data security and difference expansion algorithm for key security. Further, to reduce the encryption time of data encryption, software optimization algorithms used. Next, the encrypted data bits read, and a difference expansion algorithm is applied on it to hide the key bits in the encrypted data. The advantage of the difference expansion algorithm is that the original encrypted data bits recover with key bits extraction. The experimental results show that the data and key encapsulation takes less execution time, provide better security as compared to the existing algorithm.

Keywords— *Smart Grid, Advanced Encryption Standard, Difference Expansion, Binary Search Algorithm.*

1. INTRODUCTION

A number of stakeholders manage smart Grid (SG), and it provides electricity to worldwide houses [1]. The communication and computation abilities are introduced by smart grids into conventional grids to be connected and smart. The conventional meters of electricity are installed with storage and processing chips to perform smart functions effectively. These smart meters are connected with the home appliances for communication purposes and at electricity companies to perform the management and generation functions to provide a high connection by the smart grid. The smart grids are divided into three categories depending upon the research and examination conducted at smart grids that is smart customer, smart generation and smart grid [2]. The instant power consumption and delivering information can be managed or monitored by networked and intelligent smart grid meters; it also helps in monitoring remote control, power usage subscription, outage management, advanced demand, usage management, mainly concerning the expense like an electrical car charging at non-working hours etc. Thus it is advantageous for power distribution and generation as well as end-users. These smart meters should be further connected to smart gas and water meters for effective management and coordination of energy utilized for green/smart homes [3]. With this the main concern for privacy and security arises as benefits introduced by smart grids [4]:

- The smart meters are manipulated by hackers for consumption expense and power used.
- The fake data of power consumed at higher scale by the cyber-terrorists for attacking power system such as overloading the nuclear plants for power.
- Controlling and teasing other electrical appliances by hacking other smart meters by attackers.
- Accessing power consumed data of the sufferer by spy communication, hacking power database of the company from where they keep an eye on daily habits, activities.

To overcome these challenges, cryptography and steganography algorithms used. The cryptography algorithm encrypts the secret data with the help of the secret key and gives cipher data in the output [5]. The whole process is known as data encapsulation. On the other side, the steganography algorithm hides the data in the cover media [6]. In the literature, 3DES, AES, Homomorphic Encryption algorithms used for data encapsulation [7-9]. Out of these algorithms, AES gives better security but due to long look-up tables takes large execution time for data encryption. On the other side, to secure the secret key AES and RSA algorithm is used that takes long execution time for key encapsulation [10-11]. Thus, we have explored the steganography algorithm for key

wrapping. In the literature, the DNA based key wrapping technique proposed in that key bits hide in the DNA sequence [12]. On the receiver side, the key bits extracted from the DNA sequence but the original DNA sequence is not recovered. Therefore, in this paper, we have designed a data and key encapsulation technique using AES and a difference expansion algorithm that takes less execution time.

The main contribution of this paper is to secure the data and the secret key to the network. To achieve this goal, AES and Difference Expansion (DE) algorithms are used. In order to reduce the time complexity of data encryption, software optimization algorithms used in the AES algorithm. The binary search software optimization algorithm is used. Thereafter, the encrypted data is used for securing the key using a difference expansion algorithm. In this algorithm, the encrypted data works as a cover media in which the secret key is hidden. The difference expansion algorithm used simpler operations (such as addition, mean, difference) for it. The benefit of the proposed algorithm is given below.

- Determining the secret key in the encrypted data is difficult.
- The encrypted data bits recovered after extracting the key bits on the receiver side.

The rest of the paper is as follows. Section II explains the proposed technique. Section III shows the experimental results. The conclusion is drawn in Section IV.

2. PROPOSED TECHNIQUE

In the proposed technique, the data and key are secured. The block diagram of the proposed technique is shown in Fig. 1. In the smart grid, the user data contains various attributes such as user ID, name, address, contact number, and electricity consumption. This data is given to the AES algorithm with the secret key. The AES algorithm encrypts the secret data and gives the encrypted data in the output known as cipher data. Thereafter, the cipher data input to the difference expansion algorithm calculates the difference between adjacent cipher values and hides the key in it using the least significant bit technique and gives cipher data' in the output.

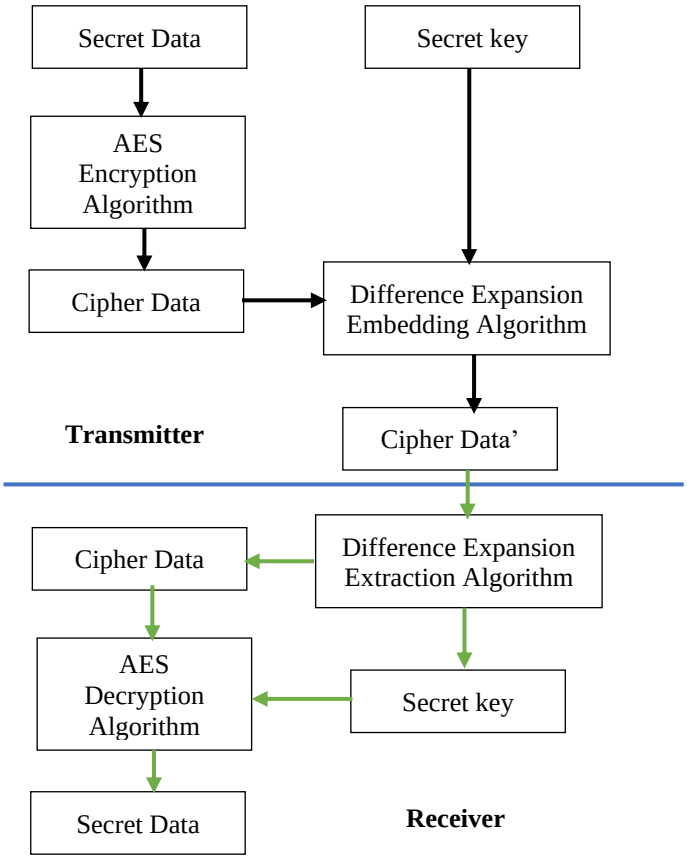

Fig. 1: Block Diagram of Proposed Technique

On the receiver side, the cipher data' is input to the difference expansion algorithm and the extraction process is performed that gives the key in the output with original cipher data bits. On the cipher data bits, the AES decryption algorithm is applied that gives the original data bits in the output. The detail description of the proposed technique is given below.

2.1 AES Algorithm

AES algorithm was designed by two cryptographers, Vincent Rijmen and Joan Daemen. AES algorithm is the symmetric algorithm. In the symmetric algorithm, the same key is used for data encryption and decryption purposes. Further, in the symmetric algorithm, it is based on the block cipher. AES has a fixed block size of 128-bit, three key size variants 128/192/ 256-bit, and a total of 10/ 12/14 rounds for data encryption and decryption purposes. The block diagram of the AES algorithm is shown in Fig. 2 [13].

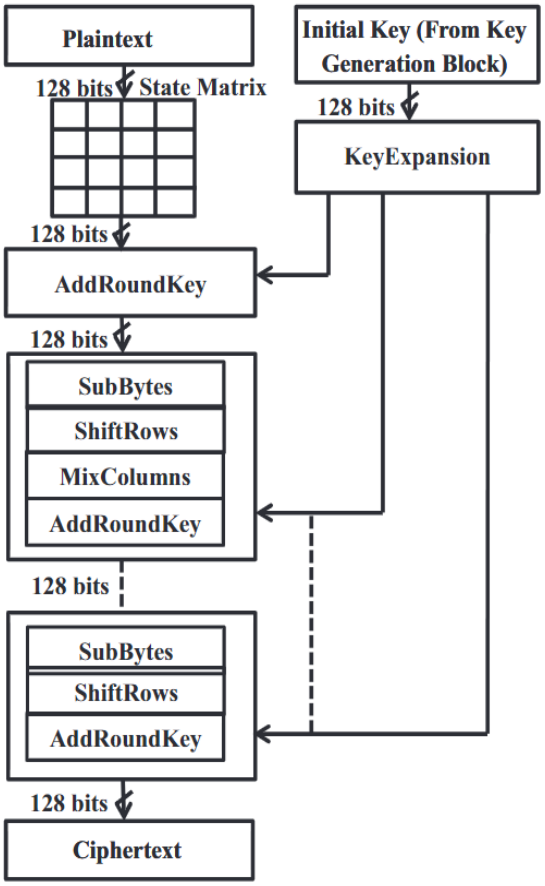

Fig. 2: Block Diagram of AES Algorithm

The plaintext and key input to the AES algorithm. The plaintext and key are $128-bit$ long. Thus, it is arranged into a $4X4$ matrix and each element of the matrix 8-bit long. Thus, the matrix element value varies from 0 to 255. The XOR operation performed between plaintext and key. Thereafter, it is passed through the substitution box known as a sub-byte stage. The substitution box substitutes the original matrix elements with the other elements as shown in Fig. 3.

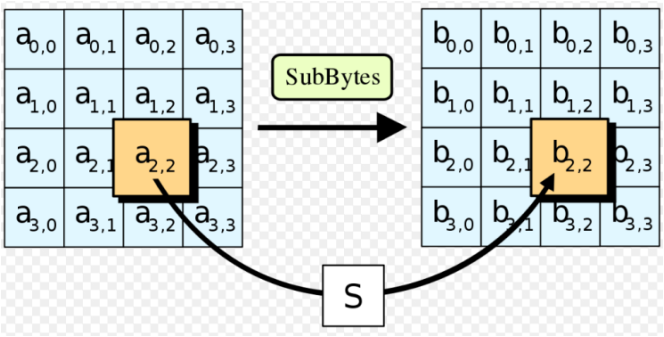

Fig. 3: Sub-Byte Step

It is a bijective mapping. The bijective mapping is one-to-one mapping and invertible function. Thus, s-box contains 2^8 combinations. These combinations are stored in the look-up table. After performing the sub-byte step, it is passed through the shift row. The shift row circulates the matrix element according to their row index value as shown in Fig. 4.

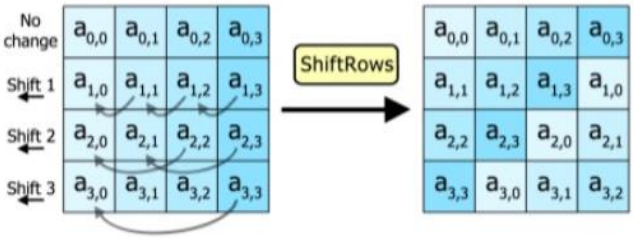

Fig. 4: Shift Row Step

The first row shifted by 0-byte, the second row shifted by 1-byte, the third row shifted by 2-byte, and the fourth row shifted by 3-byte respectively. The updated matrix is passed through the mix column step. This step performs the multiplication of the matrix with a constant matrix as shown in Fig. 5.

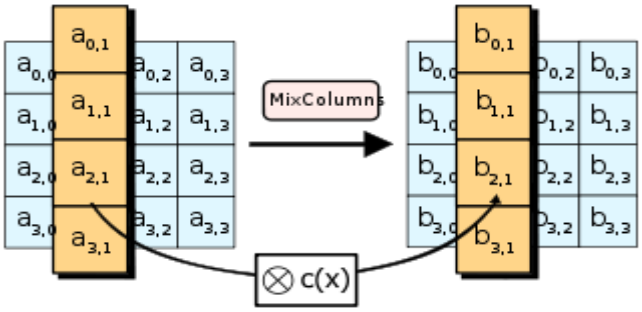

Fig. 5: Mix Column Step

The multiplication can be done directly that long execution time in each round. The optimal solution is that store the multiplication values in the look-up table. The constant matrix contains 4 numbers 2,3,1,1. Thus, multiplication value generated by multiplying with 2 and 3 numbers stored in the look-up table. The look-up table contains 2^8 combinations. In the last step, the key is updated for the next round using key expansion. The key updating process is shown in Fig. 6.

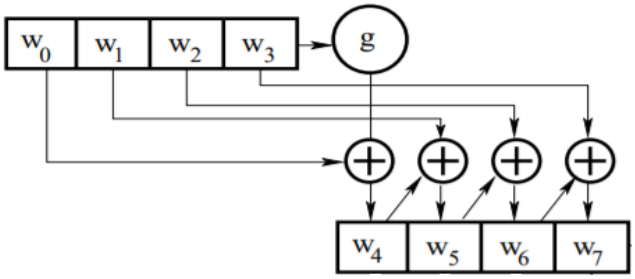

Fig. 6: Key Expansion Step

The W_0, W_1, W_2, W_3 denotes the original key values and W_4, W_5, W_6, W_7 denotes the updated key values. From the study of the AES algorithm, we found that AES contains a look-up table, and linearly searching the number takes long execution time. To reduce the execution time, we have used the binary search algorithm. The binary search algorithm reduces the searching complexity of O(n) to O($\log_2$ n). The binary search algorithm is explained below [14].

The binary search algorithm is applied to the sorted array, either ascending or descending order. In the algorithm, initially, the number is input, which we want to search in the array. After that, the binary search algorithm determines the lower and upper limit of the array and divide the array into two parts, and compared the input number with the middle value of the array. If the number is equal to the middle value, then the corresponding index is determined else number is searched in the left or right side of the middle value. In each iteration, the lower and upper limit is updated, and the array is narrow down to search the number and takes a logarithmic moment.

Example: The search list: L = 1 3 4 6 8 9 11. The found value to be: X = 4. Comparing X to 6. X is lesser. Repeating L = 1 3 4. Comparing X to 3. X is larger. Repeating with L = 4. Comparing X to 4. They are equivalent. We got X and we are done.

2.2 Difference Expansion Algorithm

The difference expansion algorithm is a steganography algorithm that hides the key in the cover media. In our work, the cipher data is worked as cover media in that the key is hidden. The difference expansion algorithm is explained with the example below [15].

The pairing host image pixels and transferring into low-pass image consisting average of integers and pixel difference consist in high-pass image is involved in DE implementation method. If pixel-pair intensity values be x and y, then l and h defined as

$$l = \left\lfloor \frac{(x+y)}{2} \right\rfloor, \qquad h = x - y \tag{1}$$

This transformation is invertible so that the gray levels x and y can be computed from l and h

$$x = l + \left\lfloor \frac{h+1}{2} \right\rfloor, \text{ and } \quad y = l - \left\lfloor \frac{h}{2} \right\rfloor \tag{2}$$

An information bit be 0 or 1, is embedded by appending it to the LSB of the h difference, thus creating a new LSB. The key bit hides using the Eq. (3).

$$h' = 2h + b \tag{3}$$

Whereas, b is the key bit.

Fig. 7 shows an example of embedding one bit in a pair of pixels. On the receiver side, the same procedure applied that gives the original pixels and key bits in the output. In the proposed algorithm, if the value of y comes negative after key hiding, then that x and y pair is not used for key hiding.

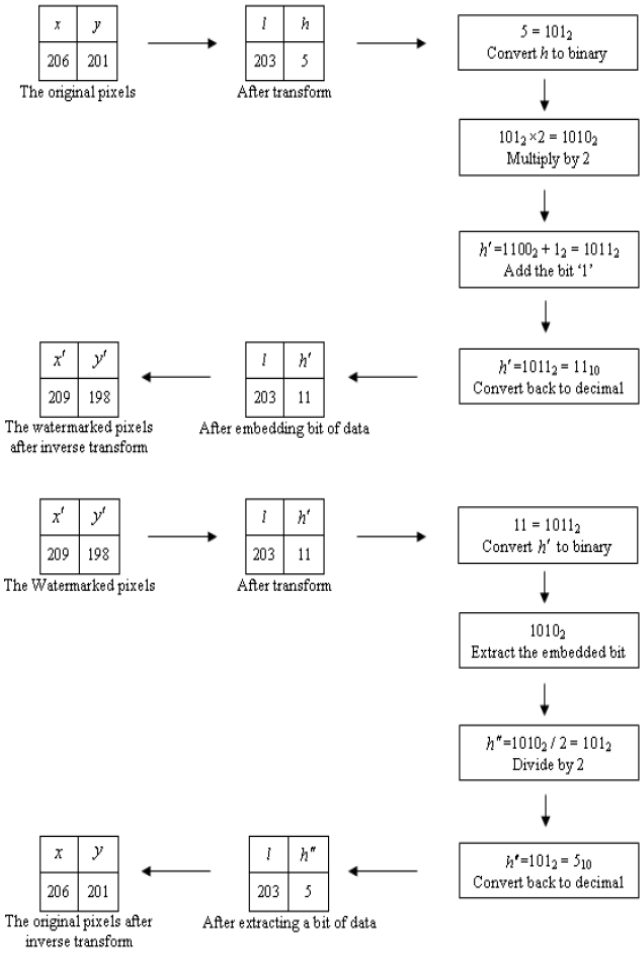

Fig. 7: Difference Expansion Algorithm

3. EXPERIMENTAL RESULTS

In this section, the experimental results of the proposed technique are explained. The algorithms are written and simulated in MATLAB 2013a. The system configuration is the i3 processor, 8GB RAM. The simulated results are shown in Table 1 is shown for the data encryption for different plaintext.

Table 1: Original and Encrypted Data

Original Data	Encrypted Data
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	70 132 152 39 250 210 122 242 183 246 47 43 105 38 252 145
49 50 51 52 97 106 97 121 57 56 53 53 50 52 32 107	39 156 251 83 37 96 82 90 214 138 190 30 41 131 177 153
0 0 0 0 0 0 0 0 56 48 56 54 97 114 116 97	163 249 12 82 107 0 211 169 12 117 221 192 165 232 226 113

The performance analysis of the proposed technique is done using various parameters. These parameters are

3.1 Execution Time

This parameter measures the total time taken by the algorithms to give the final output in the output. In the MATLAB, tic and toc commands are available to determine the execution time. This command returns the time in seconds. In the proposed technique, AES algorithm performance improved using a software optimization technique and average results are shown in Table 2.

Table 2: Comparative Analysis of Execution Time for AES and Improved AES Algorithm

Parameter	AES [13]	Optimized AES Algorithm	Total Time for Proposed Technique
Execution Time (in seconds)	12	11	12.70

3.2 Peak Signal to Noise Ratio (PSNR)

PSNR measures the ratio of the peak power to the noise power. It is measured in decibel (dB). For a good cryptography algorithm, the PSNR must below [16]. It is calculated using Eq. (4).

$$PSNR = 10 * \log_{10} (Peak^2/MSE) \quad (4)$$

whereas Peak denotes the maximum value represents can be represented in the input. Each value of the original matrix 8-bit long. Thus, the maximum value of peak can be represented in the input value is 255. MSE denotes the mean square error and it is calculated using Eq. (5).

$$MSE = \frac{1}{H \times W} \sum_{i=1}^{H} \sum_{j=1}^{W} (xij - yij)^2 \quad (5)$$

whereas HW denotes the matrix row and column. Xy denotes the original and encrypted message for cryptography. The PSNR for the proposed technique is shown in Table 3.

Table 3: PSNR for the Proposed Technique

Parameter	Proposed Technique
MSE	47.25
PSNR (in dB)	31.39

3.3 Avalanche Effect

This parameter measures the security of the algorithm. Thus, if a 1-bit change in the plaintext or key then how many bits change in the ciphertext. In the ideal case, a 50% bit change is required. It is calculated using Eq. (6).

$$Avalanche\ Effect = \frac{Number\ of\ bits\ Changed}{Length\ of\ the\ Message} X100 \quad (6)$$

In Table 5 the avalanche effect for the proposed technique is shown. The proposed technique gives a 50% avalanche effect.

Table 5: Avalanche Effect for the Proposed Technique

Parameter	Proposed Technique
Avalanche Effect	50%

4. CONCLUSION AND FUTURE WORK

In this paper, the AES and difference expansion algorithm used for data and key security. The AES algorithm execution time is reduced by deploying the binary search algorithm to search the values in the s-box lookup table. Further, the difference expansion algorithm uses simple operations for hiding the key bits in the cipher and difficult to the attacker to determine the key bits. The experimental results show that the proposed technique takes less execution time, provides better PSNR and the better avalanche effect. Thus, the proposed algorithm can be deployed for the smart grid for data and key security. In the future, other lightweight encryption algorithms and key wrapping algorithms, we shall explore.

5. REFERENCES

[1] Li, Fengjun, Bo Luo, and Peng Liu. "Secure and privacy-preserving information aggregation for smart grids." International Journal of Security and Networks 6, no. 1 (2011): 28-39.

[2] Sensing, A., Bose, A. and Wittig, W. (2008) 'Power system design: basis for efficient smart grid initiatives', IETSeminar Digests, Vol. 2008, No. 12380, pp.58–58.

[3] van Bruchem, Henk. "Think smart! The introduction of smart gas meters." In 23rd World Gas Conference, pp. 1-8. 2006.

[4] McDaniel, Patrick, and Stephen McLaughlin. "Security and privacy challenges in the smart grid." IEEE Security & Privacy 7, no. 3 (2009): 75-77.

[5] Kumari, Sarita. "A research paper on cryptography encryption and compression techniques." International Journal Of Engineering And Computer Science 6, no. 4 (2017).

[6] Liu, Yunxia, Shuyang Liu, Yonghao Wang, Hongguo Zhao, and Si Liu. "Video steganography: A review." Neurocomputing 335 (2019): 238-250.

[7] Metke, Anthony R., and Randy L. Ekl. "Security technology for smart grid networks." IEEE Transactions on Smart Grid 1, no. 1 (2010): 99-107.
[8] Li, Shaohua, KaipingXue, David SL Wei, Hao Yue, Nenghai Yu, and Peilin Hong. "SecGrid: A Secure and Efficient SGX-Enabled Smart Grid System With Rich Functionalities." IEEE Transactions on Information Forensics and Security 15 (2019): 1318-1330.
[9] Garcia, Flavio D., and Bart Jacobs. "Privacy-friendly energy-metering via homomorphic encryption." In International Workshop on Security and Trust Management, pp. 226-238. Springer, Berlin, Heidelberg, 2010.
[10] Housley, R., and M. Dworkin. Advanced Encryption Standard (AES) Key Wrap with Padding Algorithm. RFC 5649, September, 2009.
[11] Halevi, Shai, and Hugo Krawczyk. "One-pass HMQV and asymmetric key-wrapping." In International Workshop on Public Key Cryptography, pp. 317-334. Springer, Berlin, Heidelberg, 2011.
[12] Khalifa, Amal. "LSBase: A key encapsulation scheme to improve hybrid crypto-systems using DNA steganography." In 2013 8th International Conference on Computer Engineering & Systems (ICCES), pp. 105-110. IEEE, 2013.
[13] Zodpe, Harshali, and Ashok Sapkal. "An efficient AES implementation using FPGA with enhanced security features." Journal of King Saud University-Engineering Sciences (2018).
[14] AncyOommen and Chanchal Pal. "Binary Search Algorithm," International Journal of Innovative Research in Technology, vol. 1, no. 5, pp. 800-803, 2014.
[15] Al-Qershi, Osamah M., and B. E. Khoo. "An overview of reversible data hiding schemes based on difference expansion technique." In Proc. of International Conference on Software Engineering & Computer Systems (ICSECS09). 2009.
[16] Srivastava, Rupali, and O. Singh. "Performance analysis of image encryption using block based technique." International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering 4, no. 5 (2015): 4266-4271.

