Advanced Encryption Standard (AES) Algorithm to Encrypt and Decrypt Data

Ako Muhamad Abdullah

MSc Computer Science –UK

PhD Student in Computer Science

Department of Applied Mathematics & Computer Science

Eastern Mediterranean University - Cyprus

ako.abdullah@univsul.edu.iq

Student No. 16600094

Publication Date: June 16, 2017

***ABSTRACT*— Advanced Encryption Standard (AES) algorithm is one on the most common and widely symmetric block cipher algorithm used in worldwide. This algorithm has an own particular structure to encrypt and decrypt sensitive data and is applied in hardware and software all over the world. It is extremely difficult to hackers to get the real data when encrypting by AES algorithm. Till date is not any evidence to crake this algorithm. AES has the ability to deal with three different key sizes such as AES 128, 192 and 256 bit and each of this ciphers has 128 bit block size. This paper will provide an overview of AES algorithm and explain several crucial features of this algorithm in details and demonstration some previous researches that have done on it with comparing to other algorithms such as DES, 3DES, Blowfish etc.**

***Keywords*— *Cryptography, AES (Advanced Encryption Standard), Encryption, Decryption and NIST*.**

I. INTRODUCTION

Internet communication is playing the important role to transfer large amount of data in various fields. Some of data might be transmitted through insecure channel from sender to receiver. Different techniques and methods have been using by private and public sectors to protect sensitive data from intruders because of the security of electronic data is crucial issue. Cryptography is one of the most significant and popular techniques to secure the data from attackers by using two vital processes that is Encryption and Decryption. Encryption is the process of encoding data to prevent it from intruders to read the original data easily. This stage has the ability to convert the original data (Plaintext) into unreadable format known as Cipher text. The next process that has to

carry out by the authorized person is Decryption. Decryption is contrary of encryption. It is the process to convert cipher text into plain text without missing any words in the original text. To perform these process cryptography relies on mathematical calculations along with some substitutions and permutations with or without a key.

Modern cryptography provide the confidentiality, integrity, nonrepudiation and authentication [1]. These days, there are a number of algorithms have been available to encrypt and decrypt sensitive data which are typically divided into three types. Frist one is symmetric cryptography that is the same key is used for encryption and decryption data. Second one is Asymmetric cryptographic. This types of cryptography relies on two different keys for encryption and decryption. Finally, cryptographic hash function using no key instead key it is mixed the data [2].

The symmetric key is much more effective and faster than Asymmetric. Some of the common symmetric algorithms is Advance Encryption Standard (AES), Blowfish, Simplified Data Encryption Standard (S-DES) and 3DES. The main purpose of this paper will provide a detail information about Advanced Encryption Standard (AES) algorithm for encryption and decryption data then make a comparison between AES and DES algorithm to show some idea why replacing DES to AES algorithm.

This paper is organized as follows: In section 2 presents a brief history of AES algorithm. Related work discuss in section 3. In section 4 provides the evaluation criteria of AES algorithm. Basic structure of AES algorithm describe in section 5. Encryption process of AES algorithm presents in section 6. In section 7 explains the expanded key of AES. Decryption process presents in section 8. In section 9 discuss implementation areas of AES. Finally, provide a conclusion in section 10.

II. BRIEF HISTORY OF AES ALGORITHM

The Advanced Encryption Standard (AES) algorithm is one of the block cipher encryption algorithm that was published by National Institute of Standards and technology (NIST) in 2000. The main aims of this algorithm was to replace DES algorithm after appearing some vulnerable aspects of it. NIST invited experts who work on encryption and data security all over the world to introduce an innovative block cipher algorithm to encrypt and decrypt data with powerful and complex structure.

From around the world many groups submitted their algorithm. NIST accepted five algorithms for evaluate. After performing various criteria and security parameters, they selected one of the five encryption algorithm that proposed by two Belgian cryptographers Joan Daeman and Vincent Rijmen. The original name of AES algorithm is the Rijndel algorithm. However, this name has not become a popular name for this algorithm instead it is recognized as Advanced Encryption Standard (AES) algorithm around the world [14].

III. RELATED WORK

Hardware and software implementation of the AES algorithm is one of the most important area to attractive researches to do a research on it. In recent years a number of research papers have been publishing on AES algorithm to provide much more complexity and comparing the performance between the popular encryption algorithms to encrypt and decrypt data.

In [6] Lu, etal proposed a new architecture method to reduce the complexity architecture of

AES algorithm when it is implementing on the hardware such as mobile phone, PDAS and smart card etc. This method has consisted of integrating the AES encrypted and the AES decrypted to provide a perfect functional AES crypto-engine. To do that they focused on some important features of AES especially (Inv)SubBytes and (Inv)Mixcolumn module.

A study in [10] has conducted on different secret key algorithms to identify which algorithm can be provided the best performance to encrypt and decrypt data. To do that there was conducted on four common algorithms such as Blowfish, AES, DES and 3DES. In this paper to evaluate these algorithm contents and sizes of encrypting input files were changed and two different platforms were used to test these algorithms such as P-II 266 MHz and P-4 2.4 GHz. According to the results Blowfish has the ability to provide the best performance compared to other algorithms and AES has a better performance than 3DES and DES. It also provide that 3DES 1/3 throughput of DES.

In [11] provides the performance evaluation of symmetric encryption algorithms. This paper was conducted on six different common algorithms like AES, DES, 3DES, RC2, Blowfish and RC6. To compare among these algorithms different settings were performed on each algorithm such as different data types, different size of data block, different key sizes, battery power consumption and different speed for encryption and decryption data.

Under these situations there was not found significant deference when the data types were based on hexadecimal encoding or 64 encoding and there is no difference when using audio, video, text or documents. According to the results Blowfish can provide better performance compared to other algorithms when the packed size was changing, followed by RC6. On the other hand, they found that DES has high performance compared to 3DES algorithm. To time consumption RC2 provided the worst performance over all algorithms. Whereas AES has better performance than three common algorithms RC2, DES and 3DES. However, it is clear from the results when the size of key was increasing, it needs more battery and time consumption.

In this paper [14] evaluate the performance of three algorithms such as AES, DES, and RSA to encrypt text files under three parameters like computation time, memory usage, and output bytes. Encryption time was computed to convert plaintext to cipher text then comparing these algorithm to find which algorithm takes more time to encrypt text file. According to the results they have obtained RSA takes more time compared to other algorithms. For second parameters RSA needs a larger memory than AES and DES algorithms. Finally, the output byte of each algorithm has considered. DES and AES produce the same level of output byte whereas RSA has a low level of output byte.

IV. EVALUATION CRITERIA FOR AES ALGORITHM

Three important criterions were used by NIST to evaluate the algorithms that were submitted by cryptographer experts.

A. Security

One of the most crucial aspects that NIST was considered to choose algorithm it is security. The main reasons behind this was obvious because of the main aims of AES was to improve the security issue of DES algorithm. AES has the best ability to protect sensitive data from attackers and is not allowed them to break the encrypt data as compared to other proposed algorithm. This was

achieved by doing a lot of testing on AES against theoretical and practical attacks [3].

B. Cost

Another criterion that was emphasis by NIST to evaluate the algorithms it is cost. Again, the factors behind this measures was also clear due to another main purpose of AES algorithm was to improve the low performance of DES. AES was one of the algorithm which was nominated by NIST because it is able to have high computational efficiency and can be used in a wide range of applications especially in broadband links with a high speed [4].

C. Algorithm and Implementation Characteristics

This criteria was very significant to estimate the algorithms that were received from cryptographer experts. Some important aspects were measured in this stage that is the flexibility, simplicity and suitability of the algorithm for diversity of hardware and software implementation [5].

V. BASIC STRUCTURE OF AES Algorithm

AES is an iterative instead of Feistel cipher. It is based on two common techniques to encrypt and decrypt data knowns as substitution and permutation network (SPN). SPN is a number of mathematical operations that are carried out in block cipher algorithms [7]. AES has the ability to deal with 128 bits (16 bytes) as a fixed plaintext block size. These 16 bytes are represented in 4x4 matrix and AES operates on a matrix of bytes. In addition, another crucial feature in AES is number of rounds. The number of rounds is relied on the length of key. There are three different key sizes are used by AES algorithm to encrypt and decrypt data such as (128, 192 or 256 bits). The key sizes decide to the number of rounds such as AES uses 10 rounds for 128-bit keys, 12 rounds for 192-bit keys and 14 rounds for 256-bit keys [8].

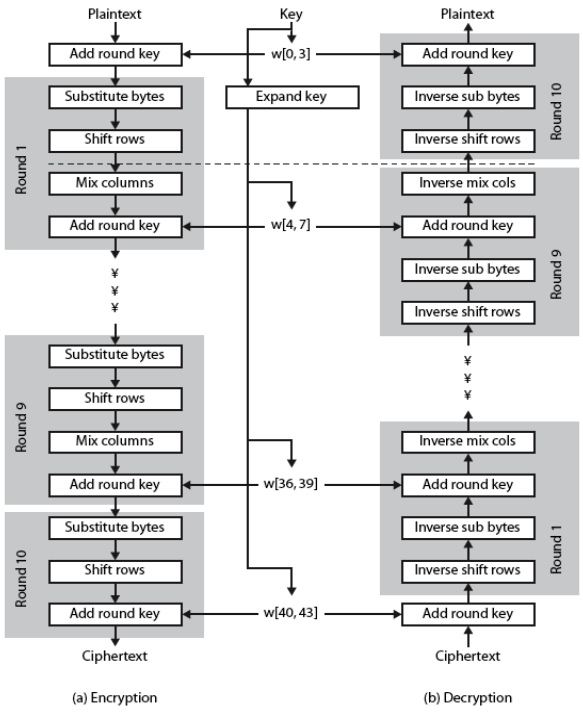

Fig. 1 Basic Structure of AES

VI. ENCRYPTION PROCESS

Encryption is a popular techniques that plays a major role to protect data from intruders. AES algorithm uses a particular structure to encrypt data to provide the best security. To do that it relies on a number of rounds and inside each round comprise of four sub-process. Each round consists of the following four steps to encrypt 128 bit block

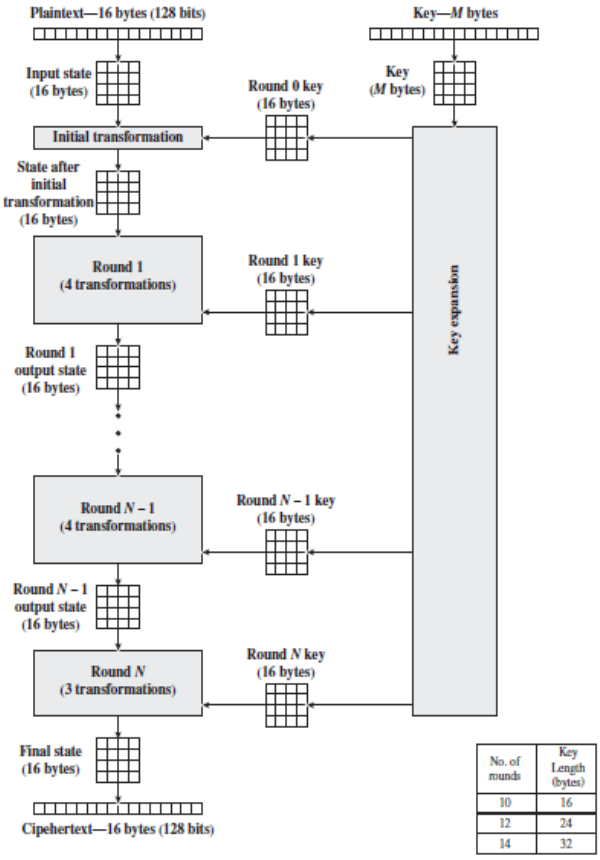

Fig.2 Encryption Processes

A. *Substitute Bytes Transformation*

The first stage of each round starts with SubBytes transformation. This stage is depends on nonlinear S-box to substitute a byte in the state to another byte. According to diffusion and confusion Shannon's principles for cryptographic algorithm design it has important roles to obtain much more security [12]. For example in AES if we have hexa 53 in the state, it has to replace to hexa ED. ED created from the intersection of 5 and 3. For remaining bytes of the state have to perform this operations.

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	63	7C	77	7B	F2	6B	6F	C5	30	01	67	2B	FE	D7	AB	76
1	CA	82	C9	7D	FA	59	47	F0	AD	D4	A2	AF	9C	A4	72	C0
2	B7	FD	93	26	36	3F	F7	CC	34	A5	E5	F1	71	D8	31	15
3	04	C7	23	C3	18	96	05	9A	07	12	80	E2	EB	27	B2	75
4	09	83	2C	1A	1B	6E	5A	A0	52	3B	D6	B3	29	E3	2F	84
5	53	D1	00	ED	20	FC	B1	5B	6A	CB	BE	39	4A	4C	58	CF
6	D0	EF	AA	FB	43	4D	33	85	45	F9	02	7F	50	3C	9F	A8
7	51	A3	40	8F	92	9D	38	F5	BC	B6	DA	21	10	FF	F3	D2
8	CD	0C	13	EC	5F	97	44	17	C4	A7	7E	3D	64	5D	19	73
9	60	81	4F	DC	22	2A	90	88	46	EE	B8	14	DE	5E	0B	DB
A	E0	32	3A	0A	49	06	24	5C	C2	D3	AC	62	91	95	E4	79
B	E7	C8	37	6D	8D	D5	4E	A9	6C	56	F4	EA	65	7A	AE	08
C	BA	78	25	2E	1C	A6	B4	C6	E8	DD	74	1F	4B	BD	8B	8A
D	70	3E	B5	66	48	03	F6	0E	61	35	57	B9	86	C1	1D	9E
E	E1	F8	98	11	69	D9	8E	94	9B	1E	87	E9	CE	55	28	DF
F	8C	A1	89	0D	BF	E6	42	68	41	99	2D	0F	B0	54	BB	16

Table 1 AES S-box Table

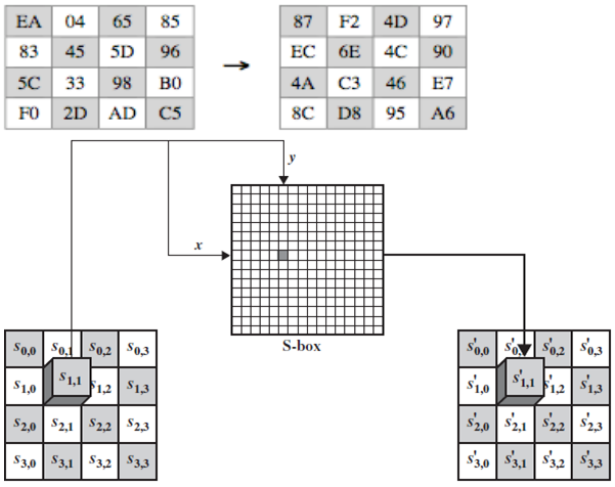

Fig. 3 Substitute byte transformation

B. *ShiftRows Transformation*

The next step after SubByte that perform on the state is ShiftRow. The main idea behind this step is to shift bytes of the state cyclically to the left in each row rather than row number zero. In this process the bytes of row number zero remains and does not carry out any permutation. In the first row only one byte is shifted circular to left. The second row is shifted two bytes to the left. The last row is shifted three bytes to the left [13]. The size of new state is not changed that remains as the same original size 16 bytes but shifted the position of the bytes in state as illustrated in Fig 4.

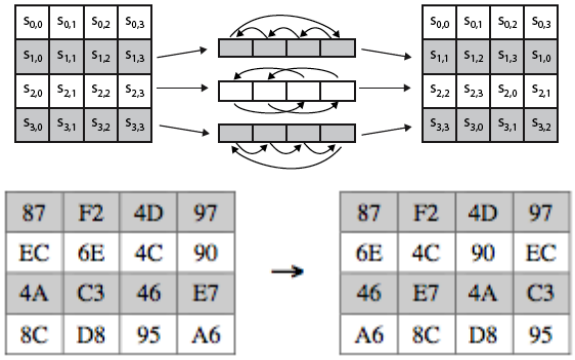

Fig.4 Shift Rows

C. MixColumns Transformation

Another crucial step occurs of the state is MixColumn. The multiplication is carried out of the state. Each byte of one row in matrix transformation multiply by each value (byte) of the state column. In another word, each row of matrix transformation must multiply by each column of the state. The results of these multiplication are used with XOR to produce a new four bytes for the next state. In this step the size of state is not changed that remained as the original size 4x4 as shown in Fig. 5.

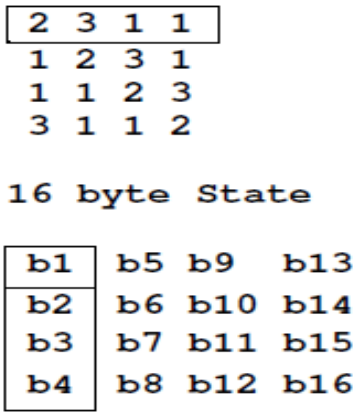

Fig. 5 Multiplication Matrix

b1 = (b1 * 2) XOR (b2 *3) XOR (b3 * 1) XOR (b4 * 1)

And so on until all columns of the state are exhausted [14].

D. AddRoundKey Transformation

AddRoundKey is the most vital stage in AES algorithm. Both the key and the input data (also referred to as the state) are structured in a 4x4 matrix of bytes [19]. Fig. 6 shows how the 128-bit key and input data are distributed into the byte matrices. AddRoundKey has the ability to provide much more security during encrypting data. This operation is based on creating the relationship between the key and the cipher text. The cipher text is coming from the previous stage. The AddRoundKey output exactly relies on the key that is indicated by users [15]. Furthermore, in the stage the subkey is also used and combined with state. The main key is used to derive the subkey in each round by using Rijndael's key schedule. The size of subkey and state is the same. The subkey is added by combining each byte of the state with the corresponding byte of the subkey using bitwise XOR [16].

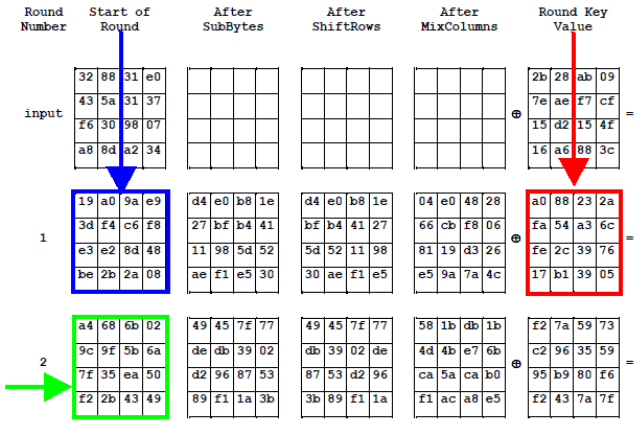

Fig. 6 Add Round Key

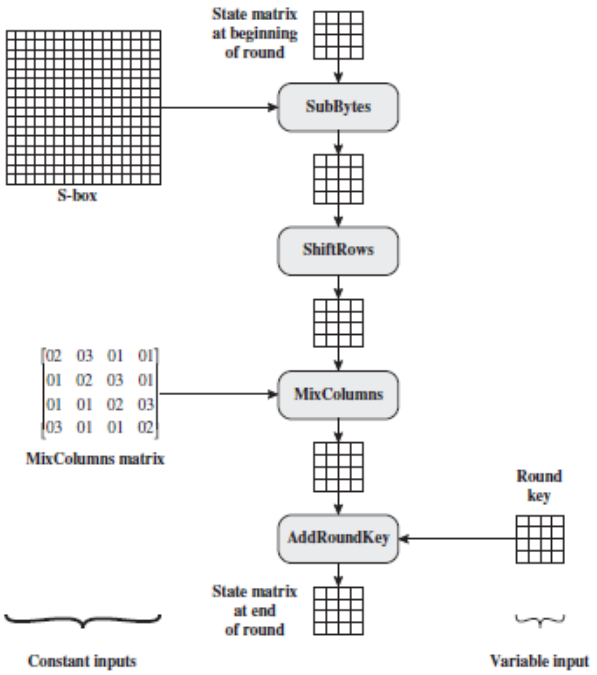

Fig.7 Inputs for Single AES Round

VII. AES KEY EXPANSION

AES algorithm is based on AES key expansion to encrypt and decrypt data. It is another most important steps in AES structure. Each round has a new key. In this section concentrates on AES Key Expansion technique. The key expansion routine creates round keys word by word, where a word is an array of four bytes. The routine creates 4x (Nr+1) words. Where Nr is the number of rounds [17]. The process is as follows:

The cipher key (initial key) is used to create the first four words. The size of key consists of 16 bytes (k0 to k15) as shown in Fig.8 that represents in an array. The first four bytes (k0 to k3) represents as w0, the next four bytes (k4 to k7) in first column represents as w1, and so on. We can use particular equation to calculate and find keys in each round easily as follows:

- **K [n]: w[i] = k [n-1]: w[i] XOR k[n]: w[i].**

This equation uses to find a key for each round rather than w0. For w0 we have to use particular equation that is different from above equation.

- **K[n]: w0 = k [n-1]: w0 XOR SubByte (k [n-1]: w3>>8) XOR Rcon [i].**

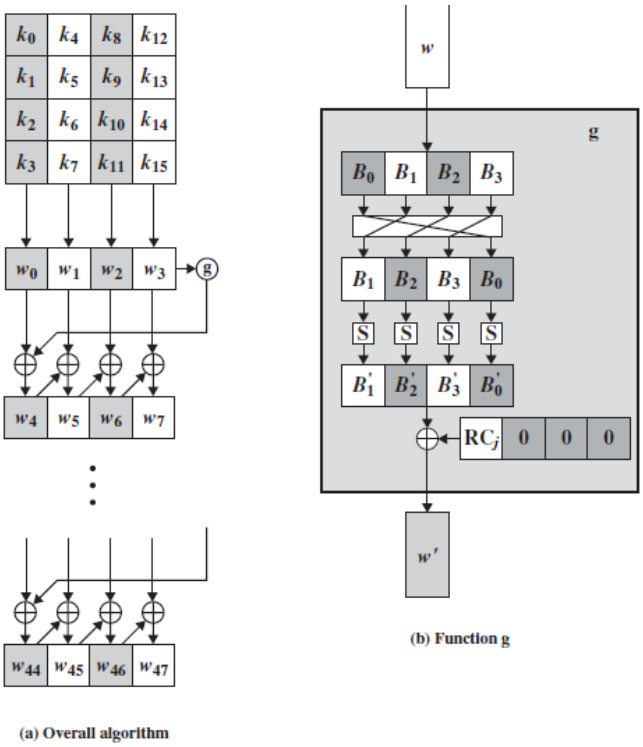

Fig. 8 AES Key Expansion

- **AES Key Expansion Example**

K1:

W0 = 0f 15 71 c9
W1 = 47 d9 e8 59
W2 = 0c b7 ad e8
W3 = af 7f 67 98

How to find K2?

K2 = w0 = k1: w0 XOR SubByte (k1:w3>>8) XOR Rcon [2]

0f 15 71 c9 XOR SubByte (af 7f 67 98>>8) XOR Rcon [2]

Rcon [2] from Auxiliary function = 02 00 00 00

0f 15 71 c9 XOR SubByte (7f 67 98 af) XOR 02 00 00 00

0f 15 71 c9 XOR D2 85 46 79 XOR 02 00 00 00

0f 15 71 c9 XOR d0 85 46 79

K2 = w0 = df q0 37 b0

K2: w1 = k1: w1 XOR k2: w0

47 d9 e8 59 XOR df q0 37 b0

K2: w1 = 98 49 df eq

K2: w2 = k1: w2 XOR k2: w1

In this example we have found W0 and W1. In a similar way we can find W2 and W3.

w0	w1	w2	w3
0f	47	0c	af
15	d9	b7	7f
71	e8	ad	67
c9	59	e8	98

W0	W1	W2	W3
df	98		
q0	49		
37	df		
b0	eq		

Fig.9 AES Key Expansion

Key Words	Auxiliary Function
w0 = 0f 15 71 c9 w1 = 47 d9 e8 59 w2 = 0c b7 ad w3 = af 7f 67 98	RotWord(w3)= 7f 67 98 af = x1 SubWord(x1)= d2 85 46 79 = y1 Rcon(1)= 01 00 00 00 y1 ⊕ Rcon(1)= d3 85 46 79 = z1
w4 = w0 ⊕ z1 = dc 90 37 b0 w5 = w4 ⊕ w1 = 9b 49 df e9 w6 = w5 ⊕ w2 = 97 fe 72 3f w7 = w6 ⊕ w3 = 38 81 15 a7	RotWord(w7)= 81 15 a7 38 = x2 SubWord(x4)= 0c 59 5c 07 = y2 Rcon(2)= 02 00 00 00 y2 ⊕ Rcon(2)= 0e 59 5c 07 = z2
w8 = w4 ⊕ z2 = d2 c9 6b b7 w9 = w8 ⊕ w5 = 49 80 b4 5e w10 = w9 ⊕ w6 = de 7e c6 61 w11 = w10 ⊕ w7 = e6 ff d3 c6	RotWord(w11)= ff d3 c6 e6 = x3 SubWord(x2)= 16 66 b4 8e = y3 Rcon(3)= 04 00 00 00 y3 ⊕ Rcon(3)= 12 66 b4 8e = z3
w12 = w8 ⊕ z3 = c0 af df 39 w13 = w12 ⊕ w9 = 89 2f 6b 67 w14 = w13 ⊕ w10 = 57 51 ad 06 w15 = w14 ⊕ w11 = b1 ae 7e c0	RotWord(w15)= ae 7e c0 b1 = x4 SubWord(x3)= e4 f3 ba c8 = y4 Rcon(4)= 08 00 00 00 y4 ⊕ Rcon(4)= ec f3 ba c8 = 4
w16 = w12 ⊕ z4 = 2c 5c 65 f1 w17 = w16 ⊕ w13 = a5 73 0e 96 w18 = w17 ⊕ w14 = f2 22 a3 90 w19 = w18 ⊕ w15 = 43 8c dd 50	RotWord(w19)= 8c dd 50 43 = x5 SubWord(x4)= 64 c1 53 1a = y5 Rcon(5)= 10 00 00 00 y5 ⊕ Rcon(5)= 74 c1 53 1a = z5
w20 = w16 ⊕ z5 = 58 9d 36 eb w21 = w20 ⊕ w17 = fd ee 38 7d w22 = w21 ⊕ w18 = 0f cc 9b ed w23 = w22 ⊕ w19 = 4c 40 46 bd	RotWord(w23)= 40 46 bd 4c = x6 SubWord(x5)= 09 5a 7a 29 = y6 Rcon(6)= 20 00 00 00 y6 ⊕ Rcon(6)= 29 5a 7a 29 = z6
w24 = w20 ⊕ z6 = 71 c7 4c c2 w25 = w24 ⊕ w21 = 8c 29 74 bf w26 = w25 ⊕ w22 = 83 e5 ef 52 w27 = w26 ⊕ w23 = cf a5 a9 ef	RotWord(w27)= a5 a9 ef cf = x7 SubWord(x6)= 06 d3 df 8a = y7 Rcon(7)= 40 00 00 00 y7 ⊕ Rcon(7)= 46 d3 df 8a = z7
w28 = w24 ⊕ z7 = 37 14 93 48 w29 = w28 ⊕ w25 = bb 3d e7 f7 w30 = w29 ⊕ w26 = 38 d8 08 a5 w31 = w30 ⊕ w27 = f7 7d a1 4a	RotWord(w31)= 7d a1 4a f7 = x8 SubWord(x7)= ff 32 d6 68 = y8 Rcon(8)= 80 00 00 00 y8 ⊕ Rcon(8)= 7f 32 d6 68 = z8
w32 = w28 ⊕ z8 = 48 26 45 20 w33 = w32 ⊕ w29 = f3 1b a2 d7 w34 = w33 ⊕ w30 = cb c3 aa 72 w35 = w34 ⊕ w32 = 3c be 0b 38	RotWord(w35)= be 0b 38 3c = x9 SubWord(x8)= ae 2b 07 eb = y9 Rcon(9)= 1B 00 00 00 y9 ⊕ Rcon(9)= b5 2b 07 eb = z9
w36 = w32 ⊕ z9 = fd 0d 42 cb w37 = w36 ⊕ w33 = 0e 16 e0 1c w38 = w37 ⊕ w34 = c5 d5 4a 6e w39 = w38 ⊕ w35 = f9 6b 41 56	RotWord(w39)= 6b 41 56 f9 = x10 SubWord(x9)= 7f 83 b1 99 = y10 Rcon(10)= 36 00 00 00 y10 ⊕ Rcon(10)= 49 83 b1 99 = z10
w40 = w36 ⊕ z10 = b4 8e f3 52 w41 = w40 ⊕ w37 = ba 98 13 4e w42 = w41 ⊕ w38 = 7f 4d 59 20 w43 = w42 ⊕ w39 = 86 26 18 76	

Fig. 10 Auxiliary Function

- **AES Encryption –Example**

To more explain the main steps of AES encryption take an example for the first round to demonstrate how to encrypt data by using AES algorithm. We have a plaintext: **AES USES A MATRIX**.

- Firstly, we have to convert this text into Hexadecimal.

Plaintext	Hexadecimal
A	00
E	04
S	12
U	14
S	12
E	04
S	12
A	00
M	0C
A	00
T	13
R	11
I	08
X	23
Z	19
Z	19

Table 2 Convert Plaintext into Hexadecimal

- Secondly, creating a matrix that based on the bytes which are obtained from above table as shown below:

00	12	0C	08
04	04	00	23
12	12	13	19
14	00	11	19

Fig. 11 State

- Thirdly, SubByte: This step relies on AES S-box but before using SubByte both the key and this matrix (also referred to as the state) are structured in a 4x4 matrix of bytes by using XOR operation as follows:

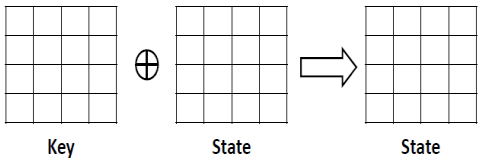

Fig. 12 Add Round Key Stage

- Second stage is ShiftRows. It has explained above. The most important stage is MixColumns. Each value in the column is eventually multiplied against every value of the matrix in a particular field (Galois Field).

63	C9	FE	30		02	03	01	01
F2	63	26	F2	X	01	02	03	01
7D	D4	C9	C9		01	01	02	03
D4	FA	63	82		03	01	01	02

Fig. 13 Multiply two States

Calculate:

63 * 02 + F2 *03 + 7D * 01 + D4 * 01

63 * 02 = 0110 0011 *02 = 1100 0110

F2 * 03 = F2 *02 + F2 * 01

= 1111 0010 *02 = 11100101 XOR 1B = 11100101 XOR 0001 1011

F2 *02 = 1111 1111

F2 *01 = 1111 0010 * 01 = 1111 0010

F2 * 02 + F2 *01 = 0000 1101 = F2 *03

7D * 01 = 0111 1101

D4 * 01 = 1101 0100

63 * 02 + F2 *03 + 7D * 01 + D4 * 01

11000110 + 00001101 + 01111101 + 11010100 = 01100010 = 62

After computing all bytes we can obtain the state as follows. In this example we calculated only one byte of the state, remaining bytes have the same procedures.

62	02	27	26
CF	92	91	0D
0C	OC	F4	D
99	18	30	74

Fig. 14 New State

- The final steps in first round is Add Round Key. This stage creates form new state of MixColumn with 128-bits of the round key by using XOR operation in a similar way others rounds.

VIII. DECRYPTION PROCESS

The decryption is the process to obtain the original data that was encrypted. This process is based on the key that was received from the sender

of the data. The decryption processes of an AES is similar to the encryption process in the reverse order and both sender and receiver have the same key to encrypt and decrypt data. The last round of a decryption stage consists of three stages such as InvShiftRows, InvSubBytes, and AddRoundKey as illustrated in Fig. 8.

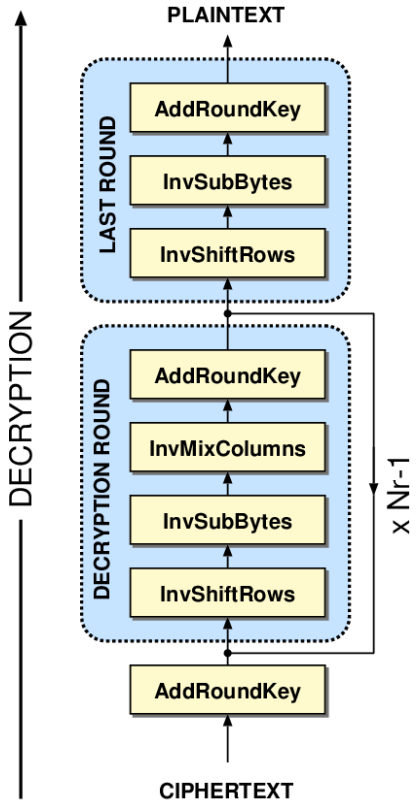

Fig. 15 Decryption Processes

IX. IMPLEMENTATION AREAS

AES algorithm is one of the most powerful algorithm that are widely used in different fields all over the world. This algorithm enables faster than DES and 3DES algorithms to encrypt and decrypt data. Furthermore, it is used in many cryptography protocols such as Socket Security Layer (SSL) and Transport Security Layer protocol to provide much more communications security between client and server over the internet. Before AES algorithm released both of protocols to encrypt and decrypt data relied on DES algorithm but after appearing some vulnerable of this algorithm the Internet Engineering Task Force (IETF) decided to replace DES to AES algorithm. AES can also be found in most modern applications and devices that need encryption functionality such as WhatsApp, Facebook Messenger and Intel and AMD processor and Cisco devices like router, switch, etc. In addition, AES Crypt package is available on many library of software programs such as C++ library, C# /.NET, Java and JavaScript which uses to easily and securely encrypt files from intruders [20].

CONCLUSION

Using internet and network are increasing rapidly. Everyday a lot of digital data have been exchanging among users. Some of data is sensitive that need to protect from intruders. Encryption algorithms play vital roles to protect original data from unauthorized access. Various kind of algorithms are exist to encrypt data. Advanced encryption standard (AES) algorithm is one of the efficient algorithm and it is widely supported and adopted on hardware and software. This algorithm enables to deal with different key sizes such as 128, 192, and 256 bits with 128 bits block cipher. In this paper, explains a number of important features of AES algorithm and presents some previous researches that have done on it to evaluate the performance of AES to encrypt data under different parameters. According to the results obtained from researches shows that AES has the ability to provide much more security compared to other algorithms like DES, 3DES etc.

REFERENCES

[1] Abdullah, A. M., & Aziz, R. H. H. (2016, June). New Approaches to Encrypt and Decrypt Data in Image using Cryptography and Steganography Algorithm., International Journal of Computer Applications, Vol. 143, No.4 (pp. 11-17).

[2] Singh, G. (2013). A study of encryption algorithms (RSA, DES, 3DES and AES) for information security. International Journal of Computer Applications, 67(19).

[3] Gaj, K., & Chodowiec, P. (2001, April). Fast implementation and fair comparison of the final candidates for Advanced Encryption Standard using Field Programmable Gate Arrays. In Cryptographers' Track at the RSA Conference (pp. 84-99). Springer Berlin Heidelberg.

[4] Stallings, W. (2006). Cryptography and network security: principles and practices. Pearson Education India.

[5] Yenuguvanilanka, J., & Elkeelany, O. (2008, April). Performance evaluation of hardware models of Advanced Encryption Standard (AES) algorithm. In Southeastcon, 2008. IEEE (pp. 222-225).

[6] Lu, C. C., & Tseng, S. Y. (2002). Integrated design of AES (Advanced Encryption Standard) encrypter and decrypter. In Application-Specific Systems, Architectures and Processors, 2002. Proceedings. The IEEE International Conference on (pp. 277-285).

[7] Mohamed, A. A., & Madian, A. H. (2010, December). A Modified Rijndael Algorithm and it's Implementation using FPGA. In Electronics, Circuits, and Systems (ICECS), 2010 17th IEEE International Conference on (pp. 335-338).

[8] Pramstaller, N., Gurkaynak, F. K., Haene, S., Kaeslin, H., Felber, N., & Fichtner, W. (2004, September). Towards an AES crypto-chip resistant to differential power analysis. In Solid-State Circuits Conference, 2004. ESSCIRC 2004. Proceeding of the 30th European IEEE (pp. 307-310).

[9] Deshpande, H. S., Karande, K. J., & Mulani, A. O. (2014, April). Efficient implementation of AES algorithm on FPGA. In Communications and Signal Processing (ICCSP), 2014 IEEE International Conference on (pp. 1895-1899).

[10] Nadeem, H (2006). A performance comparison of data encryption algorithms," *IEEE Information and Communication Technologies*, (pp. 84-89).

[11] Diaa, S., E, Hatem M. A. K., & Mohiy M. H. (2010, May) Evaluating the Performance of Symmetric Encryption Algorithms. International Journal of Network Security, Vol.10, No.3, (pp.213-219).

[12] Jain, R., Jejurkar, R., Chopade, S., Vaidya, S., & Sanap, M. (2014). AES Algorithm Using 512 Bit Key Implementation for Secure Communication. International journal of innovative Research in Computer and Communication Engineering, 2(3).

[13] Selmane, N., Guilley, S., & Danger, J. L. (2008, May). Practical setup time violation attacks on AES. In Dependable Computing Conference, 2008. EDCC 2008. Seventh European (pp. 91-96). IEEE.

[14] Berent, A. (2013). Advanced Encryption Standard by Example. Document available at URL http://www. networkdls. com/Articles/AESbyExample. pdf (April 1 2007) Accessed: June.

[15] Benvenuto, C. J. (2012). Galois field in cryptography. University of Washington.

[16] Lee, H., Lee, K., & Shin, Y. (2009). Aes implementation and performance evaluation on 8-bit microcontrollers. arXiv preprint arXiv:0911.0482.

[17] Padate, R., & Patel, A. (2014). Encryption and decryption of text using AES algorithm. International Journal of Emerging Technology and Advanced Engineering, 4(5), 54-9.

[18] Reddy, M. S., & Babu, Y. A. (2013). Evaluation of Microblaze and Implementation of AES Algorithm using Spartan-3E. International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering, 2(7), 3341-3347.

[19] Kretzschmar, U. (2009). AES128–AC Implementation for Encryption and Decryption. TI-White Paper.
[20] Wright, C. P., Dave, J., & Zadok, E. (2003, October). Cryptographic file systems performance: What you don't know can hurt you. In Security in Storage Workshop, 2003. SISW'03. Proceedings of the Second IEEE International (pp. 47-47). IEEE.