Effective Comparison and Evaluation of DES and Rijndael Algorithm (AES)

Prof.N..Penchalaiah
Department of Computer Science Engineering
ASCET
Gudur, India

Dr.R.Seshadri
Prof & Director of university computer center
S.V.University
Tirupati, India

***Abstract*: This paper discusses the effective coding of Rijndael algorithm, Advanced Encryption Standard (AES) in Hardware Description Language, Verilog. In this work we analyze the structure and design of new AES, following three criteria: a) resistance against all known attacks; b) speed and code compactness on a wide range of platforms; and c) design simplicity; as well as its similarities and dissimilarities with other symmetric ciphers. On the other side, the principal advantages of new AES with respect to DES, as well as its limitations, are investigated. Thus, for example, the fact that the new cipher and its inverse use different components, which practically eliminates the possibility for weak and semi-weak keys, as existing for DES, and the non-linearity of the key expansion, which practically eliminates the possibility of equivalent keys, are two of the principal advantages of new cipher. Finally, the implementation aspects of Rijndael cipher and its inverse are treated. Thus, although Rijndael is well suited to be implemented efficiently on a wide range of processors and in dedicated hardware, we have concentrated our study on 8-bit processors, typical for current Smart Cards and on 32-bit processors, typical for PCs.**

Keywords: Cryptography, DES, AES,Rijndael algorithm

I. INTRODUCTION

In 1997, the National Institute of Standards and Technology (NIST) initiated a process to select a symmetric-key encryption algorithm to be used to protect sensitive information in furtherance of NIST's statutory responsibilities.In 1998, NIST announced the acceptance of fifteen candidate algorithms and requested the assistance of the cryptographic research community in analyzing the candidates. This analysis included an initial examination of the security and efficiency characteristics for each algorithm. NIST reviewed the results of this preliminary research and selected MARS, RC6, Rijndael, Serpent and Twofish as finalists. An interesting performance comparison of these algorithms can be found in [3]. On October 2000 and having reviewed further public analysis of the finalists, NIST decided to propose Rijndael as the Advanced Encryption Standard (AES). Rijndael, designed by Joan Daemen (Proton World International Inc.) and Vincent Rijmen (Katholieke Univeriteit Leuven) of Belgium, is a blockcipher with a simple and elegant structure [2].

The Advanced Encryption Standard (AES), also known as the Rijndael algorithm, is a symmetric block cipher that can encrypt data blocks of 128 bits using symmetric keys of 128, 192 or 256 bits. AES was introduced to replace the Triple DES (3DES) algorithm used for a good amount of time universally. Though, if security were the only consideration, then 3DES would bean appropriate choice for a standardized encryption algorithm for decades to come. The main drawback was its slow software implementation. For reasons of both efficiency and security, a larger block size is desirable. Due to its high level security, speed, ease of implementation and flexibility, Rijndael was chosen for AES standard in the year 2001.

II RIJNDAEL ALGORITHM

The Rijndael Algorithm (pronounced "Reign Dahl," "Rain Doll" or "Rhine Dahl") is the new Advanced Encryption Standard (AES) recommended by the US National Institute of Standards and Technology (NIST) for protecting sensitive, unclassified government information. NIST has been using other encryption algorithms, such as DES (Data Encryption Standard), Triple DES and Skipjack for encrypting important government information. However, it felt in 1997 the need for a new stronger encryption algorithm to circumvent any potential threats to these algorithms from advanced hackers. Consequently, on 2 January 1997, NIST announced the initiation of the AES development effort. NIST made a formal call for algorithms on 12 September 1997. The key requirements to be fulfilled by the submitted algorithms were that they be royalty-free publicly-disclosed algorithms based on symmetric key cryptography as a block cipher and (at a minimum) support block sizes of 128-bits with key sizes of 128-bits, 192-bits and 256-bits. As a result of this call, 15 candidate algorithms from members of the cryptographic community around the globe entered the first round of scrutiny.

After evaluating these candidate algorithms in 1999, NIST selected five algorithms as the finalists. They were: MARS, RC6, Rijndael, Serpent and Twofish. These five algorithms underwent even more rigorous scrutiny in the second round of the review process. Then on 2 October 2000, NIST announced that it had selected the Rijndael algorithm as the Advanced Encryption Standard (AES). After the announcement, NIST began preparing a draft Federal Information Processing Standard (FIPS) for the AES that was finally approved as FIPS 197 in November 2001. (The US Secretary of Commerce approved the adoption of AES as an official Government standard, effective 26 May

2002.) Today, there exist four FIPSapproved encryption algorithms: AES, Triple-DES, DES and Skipjack. DES was designed by IBM and adopted by the US government as the standard encryption method for nonmilitary and non-classified use. The algorithm encrypts a 64-bit plaintext using a 56-bit key. The text is put through nineteen (19) different and very complex procedures (transposition, substitution, swapping, exclusive-OR, and rotation). Also, each step uses a different key derived from the original key and utilizes the output of the previous step as its input to create a 64-bit ciphertext. However, with the advancement in modern technology, it has now become increasingly feasible to break a DES-encrypted ciphertext. As a result, the US government came up with the Triple-DES algorithm that, as the name implies, encrypts a given plaintext by applying DES algorithm three times. If EK(I) and DK(I) represent the encryption and decryption of I using DES-key K respectively, then Triple-DES encryption O(I) is given by EK3(DK2(EK1(I))) where K1, K2, and K3 are three keys. The decryption of I using Triple-DES is given by DK1(EK2(DK3(I))). Triple-DES is backward compatible with single-DES. Thus, it is likely to remain a government standard for advanced encryption along with AES. Skipjack, another US government standard, encrypts 4-word (i.e., 8 bytes) data blocks by using permutations, exclusive-OR operations, and shifting of data in the registers for a total of 32-steps.

A combination of factors such as security, performance, efficiency, ease of implementation and flexibility contributed to the selection of this algorithm as the AES. Specifically, Rijndael appears to perform consistently well in both hardware and software platforms under a wide range of environments. These include efficient VLSI and firmware implementations in the hardware and ease of writing the code for the algorithm in various programming languages. This algorithm has excellent key setup time and good key agility. But, more importantly, without sacrificing performance, it also requires less memory for implementation. This fact makes it well suited for restricted-space environments. Furthermore, the structure of this algorithm appears to have good potential for benefiting from instruction-level parallelism.

The AES is expected to replace Triple-DES eventually because of its strong cryptographic features. The AES specifies three key sizes: 128, 192 and 256 bits. This means that, in decimal terms, there are approximately 3.4 x 1038 possible 128-bit keys, 6.2 x 1057 possible 192-bit keys, and 1.1 x 1077 possible 256-bit keys. In comparison, DES keys are 56-bits long. This bit length means that there are approximately 7.2 x 1016 possible DES keys. Thus, there are on the order of 1021 times more AES 128-bit keys than DES 56-bit keys. Assuming that one could build a machine that could recover a DES key in a second (i.e. try 255 keys per second), it would then take that machine approximately 149 thousand billion (149 trillion) years to crack a 128-bit AES key. To put that into perspective, the universe is believed to be less than 20 billion years old. With AES supporting significantly larger key sizes than what DES supports, NIST believes that this algorithm has the potential of remaining secure well beyond the next few decades.

III STRUCTURE AND DESCRIPTION OF RIJNDAEL

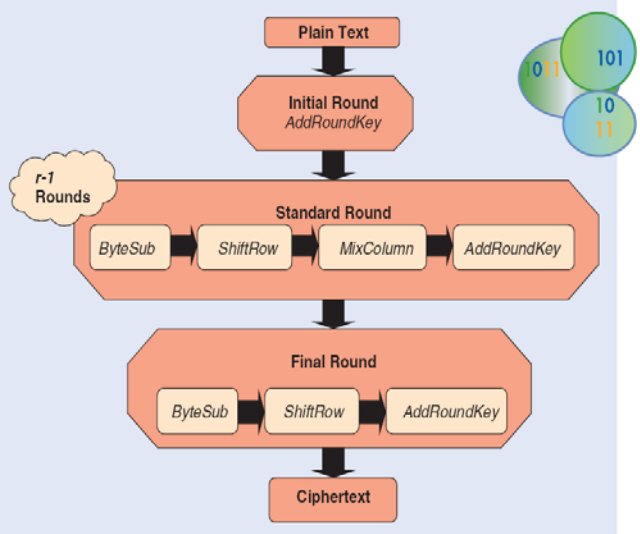

Figure 1. The Rijndael Algorithm Flowchart

A. *Mathematical preliminaries*

As we will describe, several operations in Rijndael are defined at byte level, with bytes representing elements in the Galois field $GF^{(2^8)}$. As its is known, the elements of a finite field can be represented in several ways. For any prime power there is a single finite field, hence all representations of finite field $GF^{(2^8)}$ are isomorphic [7]. Despite this equivalence, and considering the impact of the representation on the implementation complexity, the classical polynomial representation has been chosen. Thus, we can write

$$GF(2^8) = \{a_7x^7 + a_6x^6 + a_5x^5 + a_4x^4 + a_3x^3 + \\ + a_2x^2 + a_1x + a_0 \ : \ a_i \in Z_2\} \quad (1)$$

Thus, a byte consisting of bits $b_7b_6b_5b_4b_3b_2b_1b_0$ can be considered as a polynomial with coefficient in (0, l}. For example, the byte 11001010 corresponds with polynomial $x^7 + x^6 + x^3 + x$. In the polynomial representation, multiplication in $GF^{(2^8)}$ corresponds with multiplication of polynomials mod(F(z), being F(z) an irreducible polynomial of degree 8. For Rijndael,

F(z) = z8 + z4 + z3 + z + 1 (2)

Moreover, in Rijndael, other operations are defined in terms of 4-byte words. But, it is possible to define polynomials with coefficients in $GF^{(2^8)}$. In this way, a 4-byte word corresponds with a polynomial of degree below 4.

In this case, multiplication of these polynomials needs a polynomial of degree 4, in order to reduce the product to a polynomial of degree below 4. In Rijndael, this is done with the polynomial

$$M(z) = z4 + 1 \quad (3)$$

M(z) is not an irreducible polynomial over $GF(2^8)$, hence multiplication by a fixed polynomial is not necessarily invertible. In Rijndael, a fixed polynomial that does have an inverse has been chosen. As the addition in $GF(2^8)$ is the bitwise XOR, the addition of two polynomials with coefficient in this finite field is a simple bitwise XOR. However, multiplication is more complicated. Thus, assuming we have two polynomials with coefficient in $GF(2^8)$,

$$p(x) = p_3x^3 + p_2x^2 + p_1x + p_0$$
$$q(x) = q_3x^3 + q_2x^2 + q_1x + q_0 \quad (4)$$

the modular product of p (x) and q(x),

(i.e. p (x) . q(x))modM(x)), denoted by

r(x)= p (x) $\oplus$ q(x) is given by

$$r(x) = r_3x^3 + r_2x^2 + r_1x + r_0 \quad (5)$$

with

$$r_0 = p_0q_0 \oplus p_3q_1 \oplus p_2q_2 \oplus p_1q_3$$
$$r_1 = p_1q_0 \oplus p_0q_1 \oplus p_3q_2 \oplus p_2q_3$$
$$r_2 = p_2q_0 \oplus p_1q_1 \oplus p_0q_2 \oplus p_3q_3 \quad (6)$$
$$r_3 = p_3q_0 \oplus p_2q_1 \oplus p_1q_2 \oplus p_0q_3$$

or expressed as matrix multiplication

$$\begin{bmatrix} r_0 \\ r_1 \\ r_2 \\ r_3 \end{bmatrix} = \begin{bmatrix} p_0 & p_3 & p_2 & p_1 \\ p_1 & p_0 & p_3 & p_2 \\ p_2 & p_1 & p_0 & p_3 \\ p_3 & p_2 & p_1 & p_0 \end{bmatrix} \begin{bmatrix} q_0 \\ q_1 \\ q_2 \\ q_3 \end{bmatrix} \quad (7)$$

B. Structure of Rijndael

Rijndael is an iterated block cipher. It has a variable block length b and a variable key length k, which can be set to 128, 192 or 256 bits. The recommended number nr number of rounds is determined by b and k and varies between 10 and 14, as it is shown in Table 1.

TABLE 1: NUMBER OF ROUNDS (NR) AS A FUNCTION OF THE BLOCK AND KEY LENGTH.

nr	**b=4**	**b=6**	**b=8**
κ=4	10	12	14
κ=6	12	12	14
κ=8	14	14	14

In Rijndael, the State (i.e. the intermediate cipher result), *S*, can be written as a rectangular array of bytes with four rows and ***Nb*** columns, being ***Nb*** = ***LB/32,*** where ***LB*** is the block length. The cipher key is similarly written **as** a rectangular with four rows and ***Nk*** columns, being ***Nk*** = ***LK/32,*** where ***LK*** is the key length. The input and output at its external interface are considered to be one-dimensional arrays of bytes numbered upwards from 0 to ***4Nb*** - 1. The Cipher Key is also considered to be a one-dimensional array of bytes numbered upwards from 0 to ***4Nk*** - 1.

Thus, considering ***B*** the plaintext block., *K* the key and *nr* the number of rounds, we can describe the behavior of **AES as** follows:

1. Compute subkeys ***KO,*** *K1,* ..., *Kn* from the key ***K***
2. S = B $ K o
3. For ***i*** = 1 to *nr* - **3**
3.1 *S* = ByteSub(S)
3.2 *S* = ShiftRow(S)
3.3 S = MixColumn(S)
3.4 s = ***Ki*** €3
4. *S* = ByteSub(S)
5. *S* = ShiftRow(S)
6. *S* =*K* $\oplus$*S*

The inverse transformation can be described by the following steps:

1. Compute subkeys *KO,* KI, ..., *Kn* from the key *K*
2. *S*=*B*$\oplus$*K,*
3. *S* = InvShiftRow(S)
4. *S* = InvByteSub(S)
5. *S*=*Kn*$\oplus$*S*
6. For ***i*** = *nr* - **1** to **1**
3.1 *S*= *Ki*$\oplus$*S*
3.2 *S* = InvMixColumn(S)
3.3 S = InvShiftRow(S)
3.4 S = InvByteSub(S)
7. S=Ko$\oplus$S

As we can see, in the direct transformation, each round transformation is composed of four different functions, except the final round which involves only three. We briefly describe these functions and their respective inverses.

i. The ByteSub function

The function ByteSub is a nonlinear byte substitution, operating on each byte of S independently by an invertible S-box which is obtained by the composition of two transformations:

1. Each byte is represented as an element of $GF^{(2^8)}$ and substituted by its multiplicative inverse in $GF^{(2^8)}$. The value 0 is mapped onto itself.

2. Then, an affine transformation (over $GF^{(2^8)}$ defined by

$$\begin{bmatrix} y_0 \\ y_1 \\ y_2 \\ y_3 \\ y_4 \\ y_5 \\ y_6 \\ y_7 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 & 0 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 & 0 & 0 & 1 & 1 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} x_0 \\ x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \\ x_7 \end{bmatrix} + \begin{bmatrix} 1 \\ 1 \\ 0 \\ 0 \\ 0 \\ 1 \\ 1 \\ 0 \end{bmatrix} \quad (8)$$

is applied, being xo, x1, ..., x7 the bits of corresponding byte and yo, y1, ...,y7 the bits of resultant byte. The function InvByteSub is the aplication of the inverse of the corresponding S box to each byte of S.

ii. The ShiftRow function

In this function, the rows of S are cyc1icaHy shifted over different offsets. These depend on the block length Nb as we show in Table 2

TABLE 2: SHIP OFFSETS FOR DIFFERENT BLOCK LENGTHS

Nb	4	6	8
Row 0	0	0	0
Row 1	1	2	3
Row 2	1	2	3
Row 3	3	3	4

The InvShiftRow function is a cyclic shift of the rows of *S* the same number of positions, but on the left.

iii. The MixColumn function

In this function, the columns of S are considered as polynomials over GF(2') and multiplied mod(M(x)), being M (z) the polynomial given in (3), with a fixed ploynomial ~ (zg)iv en by

$$c(x) = \text{'03'}x^3 + \text{'01'}x^2 + \text{'01'}x + \text{'02'}, \quad (9)$$

where '03', '01' and '02' express hexadecimal values corresponding to x + l, l and x, respectively.

In the InvMixColumn function, every column is transformed by multiplying it with the polynomial d(x) defined by

$$c(x) \otimes d(x) = \text{'01'}, \quad (10)$$

and given by

$$d(x) = \text{'0B'}x^3 + \text{'0D'}x^2 + \text{'09'}x + \text{'0E'}, \quad (11)$$

being 'OB', 'OD', '09' and 'OE' the hexadecimal values corresponding to z3 + z + 1, x3 + x2 + 1, x3 + 1 and z3 + z2 + z, respectively.

iv. The Round Key addition

In this operation a simple bitwise XOR is applied between S and Ki (being length(Ki) is equal to the block length Nb). As it is known, this operation is its own inverse.

IV. PERFORMANCE COMPARISON BETWEEN RIJNDAEL AND DES

After comparing theoretically Rijndael with DES, it is time to analyse the performance of each of the above mentioned algorithms. Due to the fact that with DES, no operations with keys longer than 128 bits are possible, the following cases have been studied:

- DES with 64-bit key, and data length of, also, 64 bits.
- DES in CBC configuration, in order to compute 128 bits of data with a 64-bit key.
- Rijndael algorithm in its simplest form: 128-bit key, 128-bit data length.

Table 1 and Table 2 shows the results obtaining ciphering and de-cyphering using multiple times(100000 for table 1 and 100 for table 2) in order to minimize the effects given by data communication , variable initialization ,etc,(all common steps among the algorithms):

TABLE 1: TIME, IN MICROSECONDS, IN AN AMD K7-700 (PER ROUND, USING 100000 ROUNDS)

	DES 64,64	DES 64,128	Rijndael 128,128
Cyphering	3.4	6.9	35.8
De-cyphering	3.5	7.0	36.0

TABLE 2: TIME, IN MILLISECONDS, IN 8051 MICROCONTROLLER (PER ROUND, USING 100 ROUND

	DES 64,64	DES 64,128	Rijndael 128,128
Cyphering	2.8	6.1	28.8
De-cyphering	2.7	6.0	28.0

V. CONCLUSION:

In this paper, the structure and design of Rijndael cipher (new AES) have been analyzed, remarking its main advantages and limitations, as well **as** its similarities and dissimilarities with DES. Thus, the fact that the new cipher and its inverse use different components, which practically eliminates the possibility for weak and semi-weak keys, is one of the principal advantages of this new cipher algorithm, compared to DES. Also, the nonlinearity of the key expansion, which practically eliminates the possibility of equivalent keys, is another big advantage. The importance of the Advanced Encryption Standard and the high security of the Rijndael algorithm has been examined. It is learnt that Rijndael AES, at the moment is an unbreakable algorithm. With the present slow computation machines, it is really hard to break Rijndael. AES has been implemented in a large variety of languages and software tools Some code optimizations are suggested for creation of S-box and inverse mix columns transformation. It is found that the simple transformations of AES can quite comfortably implemented in any high
level or low level languages and software tools. Finally, a performance comparison among new **AES** and DES for differents microcontrollers has been carried out, showing that new AES have a computer cost of the same order

REFERENCES

[1] National Institute of Standards and Technology, Data Encryption Standard, FIPS 46-2, 1993.

[2] J. Daemen and V. Rijmen, AES Proposal: Rijndael, version 2, 1999. Available from URL: http://www. Esat. kuleuven.ac.be/vijmen/rijndael

[3] B. Schneier and D. Whiting, A Performance Comparison of the Five AES Finalist, 15 March 2000.

[4] J. Daemen and V. Rijmen, The Design of Rijndael, published by Springer-Verlag, 2002.

[5] AES webpage at US National Institute of Standards and Technology website: http://csrc.nist.gov/encryption/aes/

[6] The Rijndael's Algorithm URL: http://www.esat.kuleuven.ac.be/~rijmen/rijndael/

[7] N. Koblitz, A Course in Number Theory and Cryptography, Springer-Verlag New York Inc., 1987.

[8] M. Matsui, Linear Cryptanalysis method for DES cipher, Advances in Cryptology, Proc. Eurocrypt' 93, LNCS 765, Springer-Verlag, 1994, pp. 386- 397.

[9] S. Murphy and M. Robshaw, New observations on Rijndael, version of August 7, 2000. Available from URL: http://isg.rhbnc. ac. uk/inrobshaw.

Prof.N.Penchalaiah Research Scholar in SV University, Tirupati and Working as Professor in CSE Dept,ASCET,Gudur. He was completed his M.Tech in Sathyabama University in 2006. He has 10 years of teaching experience. He guided PG & UG Projects. He published 2 National and 1 Inter National Conferences.

Dr.R.Seshadri working as Professor & Director, University Computer Centre, Sri Venkateswara University, Tirupati. He was completed his PhD in S.V.University in 1998 in the field of " Simulation Modeling & Compression of E.C.G. Data Signals (Data compression Techniques) Electronics & Communication Engg.". He has richest of knowledge in Research field, he is guiding 10 Ph.D in Fulltime as well as Part time. He has vast experience in teaching of 26 years. He published 10 national and international conferences and 8 papers published different Journals.