Development of AES with Permutation of DES

Ahmed M. chayed
Baghdad University /mass communication collage

المستخلص :

إنّ خوارزمية تشفير AES هي من نوع تشفير الكتلةِ الذي يَستعملُ مفتاحَ تشفير وعِدّة دورات للتشفير. ان تشفير الكتلةِ هي خوارزميةُ تشفير والتي تَعْملُ على كتلة واحدة مِنْ البياناتِ في وقت واحد. في حالة تشفير AES القياسي، تكون طول الكتلة ١٢٨ قطعةً، أو ١٦ بايت. يُشيرُ تعبيرَ الدورة إلى الطريق التي يخلط فيها خوارزمية تشفيرَ البياناتِ وتقوم بإعادة التَشْفير عشَر إلى أربع عشْرة مرةِ إعتِماداً على طولِ المفتاح. ان خوارزمية AES نفسها ليستْ برنامج حاسوبِ أو نص حاسوبِ الأصلي. بل هي وصف رياضي لعملية معالجة البياناتِ. عدد مِنْ الناس خَلقوا تطبيقاتَ نص أصلي مِنْ تشفير AES، بضمن ذلك المُؤلفين الأصليين.

في هذه الورقةِ تُقدّمُ دراسة لتطوير خوارزمية AES وذلك باضافة التوزيع او نشر البيانات قبل الدخول الى الخوارزمية مما يزيد من الامنية ويمكن الرجوع الى الاصل باضافة معكوس التوزيع بعد نهاية فك الشفرة.

Abstract:

The Advanced Encryption Standard (AES) encryption algorithm is a block cipher that uses an encryption key and a several rounds of encryption. A block cipher is an encryption algorithm that works on a single block of data at a time. In the case of standard AES encryption the block is 128 bits, or 16 bytes, in length. The term "rounds" refers to the way in which the encryption algorithm mixes the data re-encrypting it ten to fourteen times depending on the length of the key[1]. The AES algorithm itself is not a computer program or computer source code. It is a mathematical description of a process of obscuring data. A number

of people have created source code implementations of AES encryption, including the original authors.

In this paper we introduce a development study for the AES with new permutation of portion. Since we add initial permutation to beginning of AES encryption and when decrypt add permutation inverse to its end. These make it more confusion then orginal AES.

1-Introduction

Secrecy is the heart of cryptography. Encryption is a practical means to achieve information secrecy. Modern encryption techniques are mathematical transformations (algorithms) which treat messages as numbers or algebraic elements in a space and transform them between a region of "meaningful messages" and a region of "unintelligible messages". In order to restore information, an encryption transformation must be reversible and the reversing transformation is called decryption. Conventionally, encryption and decryption algorithms are parameterized by cryptographic keys. An encryption algorithm and a decryption algorithm plus the description on the format of messages and keys form a cryptographic system or a cryptosystem[2].

Semantically, Shannon characterizes a desired property for a cryptosystem as follows: the cipher text message space is the space of all possible messages while the clear text message space is a sparse region inside the message space, in which messages have a certain fairly simple statistical structure, i.e., they are meaningful; a (good) encryption algorithm is a mixing transformation which distributes the meaningful messages from the sparse and meaningful region fairly uniformly over the entire message space[1].

2-Cryptography

As the scope of cryptography has broadened in recent years attempts have been made to lay more rigorous mathematical foundations for the subject. While cryptography has historically been seen as an art rather than a science this has always really depended on which side of the 'cryptographic fence' you belong. We distinguish between *cryptographers*, whose job it is to design cryptographic systems, and *cryptanalysts*, whose job it is to try to break them. Cryptanalysts have been using mathematics to break ciphers for more than a thousand years.

One could argue that cryptographers have been less scientific when designing cryptosystems. They have often relied on intuition to guide their choice of cipher. A common mistake that is repeated throughout the history of cryptography is that a 'complicated' cryptosystem must be secure. As we will see those cryptosystems which are currently believed to be most secure are really quite simple to describe[3].

The massive increase in the public use of cryptography, driven partly by the advent of the Internet, has led to a large amount of work attempting to put cryptography on a firm scientific footing. In many ways this has been extremely successful: for example it is now possible to agree (up to a point) on what it means to say that a cryptographic protocol is secure. However, we must caution against complacency: the inability to prove that certain computational problems are indeed 'difficult' means that almost every aspect of modern cryptography relies on extremely plausible, but nevertheless unproven, security assumptions [2]. In this respect modern cryptography shares some unfortunate similarities with the cryptography of earlier times!

3-Block Cipher Principles

Most symmetric block encryption algorithms in current use are based on a structure referred to as a Feistel block cipher. For that reason, it is important to examine the design principles of the Feistel cipher. We begin with a comparison of stream ciphers and block ciphers. Then we discuss the motivation for the Feistel block cipher structure. Finally, we discuss some of its implications[4].

The most widely used encryption scheme is based on the Data Encryption Standard (DES) adopted in 1977 by the National Bureau of Standards, now the National Institute of Standards and Technology (NIST), as Federal Information Processing Standard 46 (FIPS PUB 46). The algorithm itself is referred to as the Data Encryption Algorithm (DEA). For DES, data are encrypted in 64-bit blocks using a 56-bit key. The algorithm transforms 64-bit input in a series of steps into a 64-bit output. The same steps, with the same key, are used to reverse the encryption[5].

The terminology is a bit confusing. Until recently, the terms DES and DEA could be used interchangeably. However, the most recent edition of the DES document includes a specification of the DEA described here plus the triple DEA (TDEA). Both DEA and TDEA are part of the Data Encryption Standard. Further, until the recent adoption of the official term TDEA, the triple DEA algorithm was typically referred to as triple DES and written as 3DES. For the sake of convenience, we use the term 3DES[4].

4- Advance Encryption Standard (AES)

The National Institute of Standards and Technology (NIST) mentioned that in 1999, issued a new version of its DES standard (FIPS

PUB 46-3) (FIPS stands for Federal Information Processing Standard.) that indicated that DES should only be used for legacy systems and that triple DES (3DES) be used. 3DES has two attractions that assure its widespread use over the next few years. First, with its 168-bit key length, it overcomes the vulnerability to brute-force attack of DES. Second, the underlying encryption algorithm in 3DES is the same as in DES. This algorithm has been subjected to more security than any other encryption algorithm over a longer period of time, and no effective cryptanalytic attack based on the algorithm rather than brute force has been found. Accordingly, there is a high level of confidence that 3DES is very resistant to cryptanalysis. If security were the only consideration, then 3DES would be an appropriate choice for a standardized encryption algorithm for decades to come.

The principal drawback of 3DES is that the algorithm is relatively sluggish in software. The original DES was designed for mid-1970s hardware implementation and does not produce efficient software code. 3DES, which has three times as many rounds as DES, is correspondingly slower. A secondary drawback is that both DES and 3DES use a 64-bit block size. For reasons of both efficiency and security, a larger block size is desirable[1][2].

Because of these drawbacks, 3DES is not a reasonable candidate for long-term use. As a replacement, NIST in 1997 issued a call for proposals for a new Advanced Encryption Standard (AES), which should have a security strength equal to or better than 3DES and significantly improved efficiency. In addition to these general requirements, NIST specified that AES must be a symmetric block cipher with a block length of 128 bits and support for key lengths of 128, 192, and 256 bits.

In a first round of evaluation, 15 proposed algorithms were accepted. A second round narrowed the field to 5 algorithms. NIST completed its evaluation process and published a final standard (FIPS PUB 197) in November of 2001. NIST selected Rijndael as the proposed AES algorithm. The two researchers who developed and submitted Rijndael for the AES are both cryptographers from Belgium: Dr. Joan Daemen and Dr. Vincent Rijmen[6].

Ultimately, AES is intended to replace 3DES, but this process will take a number of years. NIST anticipates that 3DES will remain an approved algorithm (for U.S. government use) for the foreseeable future[7].

Like the DES (and most modern symmetric-key block ciphers), the Rijndael algorithm comprises a plural number of iterations of a basic unit of transformation: "round." In the minimum case of 128-bit message-block and key-block size, the number of rounds is 10. For larger message sizes and key sizes, the number of rounds should be increased accordingly [8].

A round transformation in Rijndael is denoted by:

Round (State, RoundKey)

Here State is a round-message matrix and is treated as both input and output; RoundKey is a round-key matrix and is derived from the input key via key schedule. The execution of a round will cause the elements of State to change value (i.e., to change its state). For encryption (respectively, decryption), State input to the first round is Input Block which is the plaintext (respectively, ciphertext) message matrix, and State output from the final round is the ciphertext (respectively, plaintext) message matrix[9] see figure 1.

High-level description of the AES algorithm

Input :plaintext

Output :ciphertext

- KeyExpansion using Rijndael's key schedule
- Initial Round

1. AddRoundKey

- Rounds

1. SubBytes—a non-linear substitution step where each byte is replaced with another according to a lookup table.
2. ShiftRows—a transposition step where each row of the state is shifted cyclically a certain number of steps.
3. MixColumns—a mixing operation which operates on the columns of the state, combining the four bytes in each column
4. AddRoundKey—each byte of the state is combined with the round key; each round key is derived from the cipher key using a key schedule[6].

- Final Round (no MixColumns)

1. SubBytes
2. ShiftRows
3. AddRoundKey

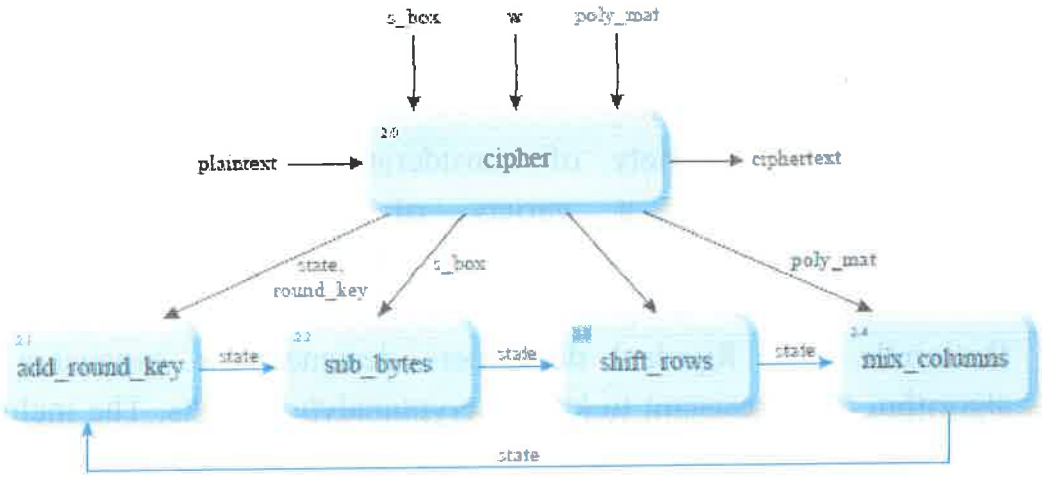

Figure 1 AES Encryption function cipher

4-1 AES Evaluation

It is worth examining the criteria used by NIST to evaluate potential candidates. These criteria span the range of concerns for the practical application of modern symmetric block ciphers. In fact, two set of criteria evolved. When NIST issued its original request for candidate algorithm nominations in 1997 [10], The three categories of criteria were as follows:

- Security: This refers to the effort required to cryptanalyze an algorithm. The emphasis in the evaluation was on the practicality of the attack. Because the minimum key size for AES is 128 bits, brute-force attacks with current and projected technology were considered impractical. Therefore, the emphasis, with respect to this point, is cryptanalysis other than a brute-force attack.
- Cost: NIST intends AES to be practical in a wide range of applications. Accordingly, AES must have high computational

efficiency, so as to be usable in high-speed applications, such as broadband links.

- Algorithm and implementation characteristics: This category includes a variety of considerations, including flexibility; suitability for a variety of hardware and software implementations; and simplicity, which will make an analysis of security more straightforward[13].

Rationale: The Rijndael developers designed the expansion key algorithm to be resistant to known cryptanalytic attacks. The inclusion of a round-dependent round constant eliminates the symmetry, or similarity, between the ways in which round keys are generated in different rounds. The specific criteria that were used are as follows [11]:

- Knowledge of a part of the cipher key or round key does not enable calculation of many other round key bits
- An invertible transformation [i.e., knowledge of any Nk consecutive words of the Expanded Key enables regeneration the entire expanded key (Nk = key size in words)]
- Speed on a wide range of processors
- Usage of round constants to eliminate symmetries
- Diffusion of cipher key differences into the round keys; that is, each key bit affects many round key bits
- Enough nonlinearity to prohibit the full determination of round key differences from cipher key differences only
- Simplicity of description

The idea is that if you know less than Nk consecutive words of either the cipher key or one of the round keys, then it is difficult to reconstruct the remaining unknown bits. The fewer bits one knows, the more

difficult it is to do the reconstruction or to determine other bits in the key expansion [12].

5-The Proposed Developed AES with Permutation

The initial permutation and its inverse are defined by tables, as shown in Tables 1a and 1b, respectively. These tables are used to increase the confusion for the plaintext and illustrated as follows. The input to a table consists of 128 bits numbered from 1 to 128 The 128 entries in the permutation table contain a permutation of the numbers from 1 to 128. Each entry in the permutation table indicates the position of a numbered input bit in the output, which also consists of 128 bits. If we then take the inverse permutation $Y = IP^{-1}(X) = IP^{-1}(IP(M))$, it can be seen that the original ordering of the bits is restored.

Table 1 Permutation Tables for proposed system

(a) Initial Permutation (IP)

114	98	82	66	50	34	18	2
116	100	84	68	52	36	20	4
118	102	86	70	54	38	22	6
120	104	88	72	56	40	24	8
122	106	90	74	58	42	26	10
124	108	92	76	60	44	28	12
126	110	94	78	62	46	30	14
128	112	96	80	64	48	32	16
113	97	81	65	49	33	17	1
115	99	83	67	51	35	19	3

Table 1 Permutation Tables for proposed system

(a) Initial Permutation (IP)

117	101	85	69	53	37	21	5
119	103	87	71	55	39	23	7
121	105	89	73	57	41	25	9
123	107	91	75	59	43	27	11
125	109	93	77	61	45	29	13
127	111	95	79	63	47	31	15

(b) Inverse Initial Permutation (IP^{-1})

80	16	96	32	112	48	128	64
79	15	95	31	111	47	127	63
78	14	94	30	110	46	126	62
77	13	93	29	109	45	125	61
76	12	92	28	108	44	124	60
75	11	91	27	107	43	123	59
74	10	90	26	106	42	122	58
73	9	89	25	105	41	121	57
72	8	88	24	104	40	120	56
71	7	87	23	103	39	119	55
70	6	86	22	102	38	118	54
69	5	85	21	101	37	117	53
68	4	84	20	100	36	116	52
67	3	83	19	99	35	115	51
66	2	82	18	98	34	114	50

65	1	81	17	97	33	113	49

The developing of AES is to add the initial permutation in the beginning of encryption and inverse permutation in the end of decryption as in the figure 2 below.

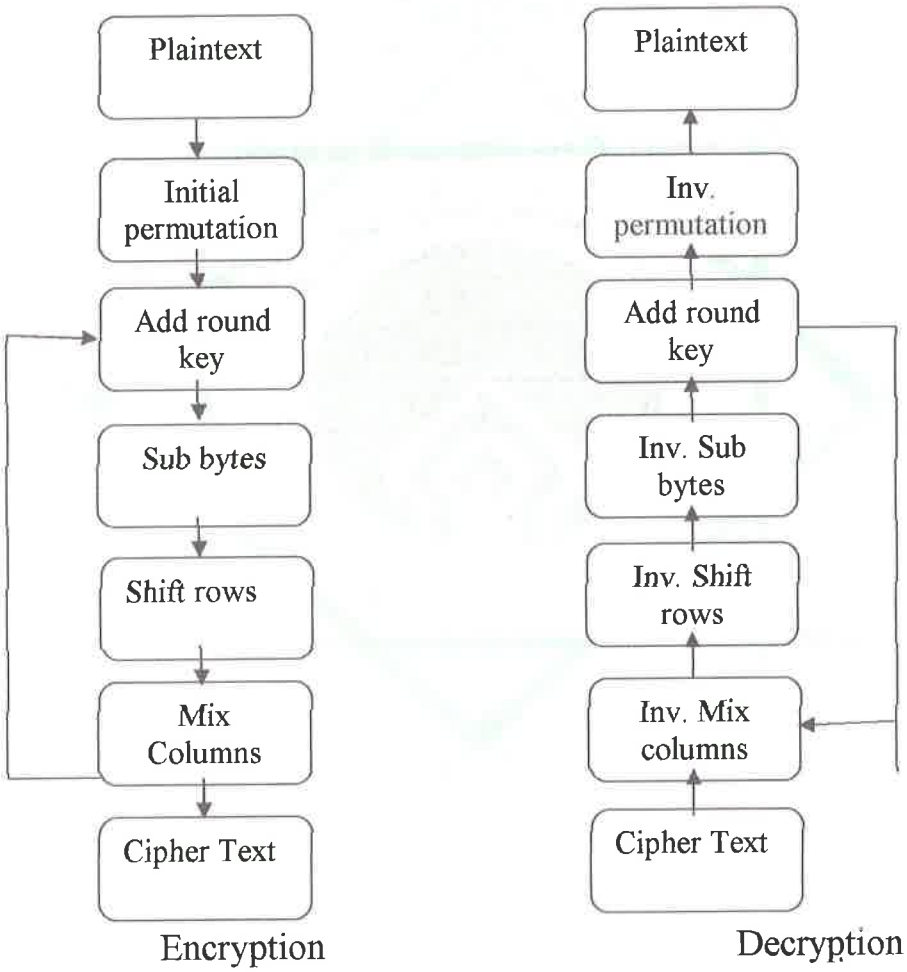

Figure 2 Developed AES Encryption/Decryption Model

6-Implementation

The proposed developed AES model is implemented in C++programming language. Where we take various plaintext messages with different sizes. The following table explains the run time execution for the proposed model compared to the original AES model.

Table 2 The Running Time measured in msec.

Test no.	Plaintext Size	AES	Developed AES
1	10 kb	0.023	0.025
2	100 kb	0.2	0.21
3	1 Mb	1.9	2.0
4	10 Mb	17.7	17.9

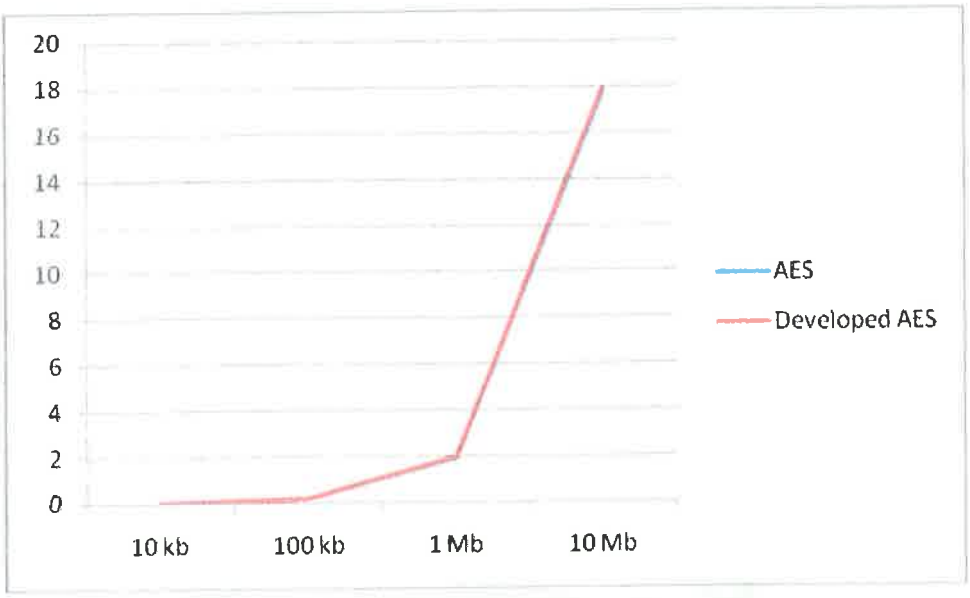

Figure 3 The Running Time Execution

Figure 3 present that the original AES and the development AES have the same time of execution.

7- Result and discussion

While there is much debate about the security and performance of AES, there is a consensus that it is significantly more secure than even 3DES, and in many environments faster. The AES is available in three key sizes: 128, 192 and 256 bits, versus the 56 bit DES. Therefore, there are around 1021 times more AES 128-bit keys than DES 56-bit keys.

Some cryptanalysts have also suggested that AES performance is up to 40% faster in hardware and software than 3DES, although it's open to debate and interpretation.

As one expert put it, assuming that you could recover a DES key in a second (trying 255 keys per second), it would take the same machine

approximately 149 trillion years to recover a 128-bit AES key. The conclusion can be summarized in table 3 below:

Table 3 comparison between AES, DES, 3DES and the proposed developed AES

	AES	TDES/ 3DES	DES	Developed AES
Description	Advanced Encryption Standard	Triple Data Encryption Standard	Data Encryption Standard	Developed Advanced Encryption Standard
Timeline	Official standard since 2001	Standardized 1977	1977	Proposed in this paper
Type of algorithm	Symmetric block cipher	Symmetric block cipher	Symmetric block cipher	Symmetric block cipher
Block size	128, 192, or 256 bits	64 bits	64 bits	128, 192, or 256 bits
Key size (in bits)	Variable (128, 192, or 256)	Fixed(168)	Fixed(56)	Variable (128, 192, or 256)
Speed	High	Low	Medium	High
Time to crack (assume a machine *could try* 255 keys per second - NIST)	149 trillion years	4.6 billion years	400 days	More than 149 trillion years
Resource consumption	Low	Medium	Medium	Low

Style Structure	No Fiestal fashion based	Fiestal fashion based	Fiestal fashion based	No Fiestal fashion based
Algorithm behavior	Not Invertible algorithm	Invertible algorithm	Invertible algorithm	Not Invertible algorithm
Number of round	10,12, 14depend of key size	48	16	10,12, 14depend of key size
Algorithm style	Strength Algebraic based	Boolean Algebra	Boolean Algebra	Strength Algebraic based
Cryptanalysis resistance	Strong against differential, truncated differential, linear, interpolation and Square attacks	Vulnerable to differential and linear cryptanalysis; weak substitution tables	Vulnerable to differential and linear cryptanalysis; weak substitution tables	Strong against differential, truncated differential, linear, interpolation and Square attacks
Security	Considered secure	Proven inadequate	Proven inadequate	Considered secure

8- Conclusion

The Proposed AES proves to be better security than DES, Triple DES and AES, as it takes considerably much more time to break by the brute force program for a given key length.

Time Taken to break Proposed AES algorithm by a brute force program increases exponentially with increase in the key lengths. The developed method it make more secrecy getting from the additional permutation

for the input plaintext with add very small time for encryption and decryption. But it increase time of attack.

References

1- Shafi Goldwasser," Lecture Notes on Cryptography", Cambridge university press,2001

2- Wenbo Mao," Modern Cryptography: Theory and Practice", Prentice Hall PTR,2003

3- Menezes, A. J., van Oorschot, P. C. and Vanstone, S. A. "Handbook of Applied
Cryptography". Boca Raton, New York, London and Tokyo, CRC Press. (1996).

4-William Stallings," Cryptography and Network Security Principles and Practices", Fourth Edition, Prentice Hall,2005

5- *john T. ,and Dominic W.," Complexity and cryptography* An Introduction", Cambridge university press,2006

6- Murphy, S. and Robshaw, M. (2002). Essential algebraic structure within the AES. In Advances in Cryptology CRYPTO 2002, Springer-Verlag Lecture Notes in Computer Science, 2442, 1–16.

7- *Daemen & Rijmen* ,"*The Rijndael* AES Proposal", http://www.nist.gov/CryptoToolkit , 1998

8- J.P. BUHLER, H.W. LENSTRA JR., AND C. POMERANCE, "Factoring integers with the number field sieve", A.K. Lenstra and H.W. Lenstra Jr., editors, The Development of the Number Field Sieve, volume 1554 of Lecture
Notes in Mathematics, 50–94, Springer- Verlag, 1993.

9- National Institute of Standards and Technology. "Request for Candidate Algorithm Nominations for the Advanced Encryption Standard." Federal Register, September, 1997.

10- Nechvatal, J., et al. Report on the Development of the Advanced Encryption Standard. National Institute of Standards and Technology. October 2, 2000.

11-Kocher, P.; Jaffe, J.; and Jun, B. "Introduction to Differential Power Analysis and Related Attacks." http://www.cryptography.com/dpa/technical/index.html,1999.

12-Biham, E., and Shamir, A. "Power Analysis of the Key Scheduling of the AES Candidates" Proceedings, Second AES Candidate Conference, 24 October 2000. http://csrc.nist.gov/encryption/aes/round1/conf2/aes2conf.htm

13-Daemen, J., and Rijmen, V. AES Proposal: Rijndael, Version 2. Submission to NIST, March 1999. http://csrc.nist.gov/encryption/aes