International Journal of Applied Engineering Research
ISSN 0973-4562 Volume 10, Number 3 (2015) pp. 6745-6754

http://www.ripublication.com

A Novel Cipher Security Mechanism For IEEE 802.11i

K.Antony Kumar[1], N.K.Manikandan[2], D.Manivannan[3] S.Saran Raj[4]

[1,2,3,4]Assistant Professor, [1,2,3,4]Department of Computer Science and Engineering,
[1,23]Vel Tech University, Chennai, India
[4]Vel Tech Multi Tech Dr.Rangarajan Dr.Sakunthala Engineering College, Chennai,

[1]antonykmr32@gmail.com , [2]manikandan1488@gmail.com,[3]mani02.ceg@gmail.com
[4]sarandillip.er@gmail.com

Abstract

In today's environment due to rapid development of internet growth, wireless security seems to be an important aspect of the communication / message transmission. Wireless security prevents unauthorized access or damage to confidential information. There are several cryptographic techniques in which the current encryption standard for wireless networks recommends the AES algorithm. In the counter mode of AES algorithm, 128 bit input data is encrypted with 128 bit key brook and produces 128 bit encrypted output data before transmission. Moreover, the length of the data is directly proportional to the energy level consumption. This problem is addressed by using a novel cipher security mechanism called Diffusioncipher. The Diffusioncipher uses the AES algorithm along with Counter (CTR) mode and Cipher Block Chaining (CBC). This Diffusion cipher securely expands the given 128 bit counter value to a larger 288 bit key brook. In order to reduce energy loss, the data block size is increased and so that the encryption per frame decreases. When Diffusion cipher is used instead of AES, we observe that energy efficiency due to Diffusion cipher is significant for larger frame lengths.

Keywords- Encryption, Decryption, Authentication, Confidentiality, Diffusioncipher.

I. INTRODUCTION

Cryptography is the science of secret codes, enabling the confidentiality of communication through an insecured channel. It protects against unauthorized parties, by preventing unauthorized alteration of use. It uses 'n' cryptographic system, to transform a plaintext into a cipher text, with the help of a key.

Encryption is of prime importance when confidential data is transmitted over the network. There is a huge amount of confusion and Diffusionof the data during encryption, which makes it very difficult for an attacker to interpret the encryption pattern, and the plaintext form of the encrypted data.

Wireless security is the prevention from unauthorized access or damage to computers using wireless networks. The most common types of wireless security are Wired Equivalent Privacy (WEP) and Wi-Fi Protected Access (WPA). WEP [3] is one of the least secure forms of security. WPA was a quick alternative to improve security over WEP. The current standard is WPA2; some hardware cannot support WPA2, without firmware upgrade o r replacement. WPA2 uses an encryption device, which encrypts the network with a 256 bit key; the longer key length improves security over WEP.

IEEE 802.1X [3] is the IEEE Standard authentication mechanism to devices wishing to attach to a Wireless LAN. The Wired Equivalent Privacy (WEP) encryption standard was the original encryption standard for wireless, but the ratification WPA2 the IEEE has declared it "deprecated", and while often supported, it is seldom or never the default on modern equipment.

The Wi-Fi Protected Access (WPA and WPA2) security protocols were created later to address the problems with WEP. If a weak password, such as a dictionary word or short character string is used, WPA and WPA2 can be cracked. Using a long enough random password (e.g. 14 random letters) or passphrase (e.g. 5 randomly chosen words) makes pre-shared key WPA virtually uncrackable. The second generation of the WPA security protocol (WPA2) is based on the final IEEE 802.11i [3] amendment to the 802.11 standard and is eligible for FIPS 140-2 compliance. With all those encryption schemes, any client in the network who knows the keys, can read all the traffic. Wi-Fi Protected Access (WPA) is a software/firmware improvement over WEP. All regular WLAN-equipment, that worked with WEP are able to be simply upgraded and no new equipment needs to be bought. WPA is a trimmed-down version of the 802.11i security standard, that was developed by the IEEE 802.11, to replace WEP. The TKIP encryption algorithm was developed for WPA, to provide improvements to WEP, that could be fielded as, firmware upgrades to existing 802.11 devices.

The current security mechanisms for the IEEE 802.11i (WPA2) makes use of the Advanced Encryption Standard (AES[7]) block cipher (based on Rijndael [8]); to provide both authentication and confidentiality in a single protocol called, Counter Mode with Cipher Block Chaining Message Authentication Code Protocol (CCMP). The IEEE 802.11i architecture [2] contains the following components: IEEE 802.1X for authentication, Robust Security Network (RSN) for keeping track of associations, and AES based CCMP to provide confidentiality, integrity and origin authentication.

The reason for the switch from the RC4 based WEP [16] and TKIP [17] (the predecessors to WPA2) to the AES based CCMP was, due to the superior security of the AES in comparison to the RC4. However, the drawback of AES- CCMP is that, it consumes more energy compared to its predecessor. This is because, the RC4 cipher used in WEP is a stream cipher; whereas, the AES used in CCMP is inherently a block cipher used in stream (counter or CTR) mode. Therefore, a full 10 round AES

needs to be performed to encrypt every 128 bits of *MPDU*. For larger frame sizes, this approach is inefficient in energy, and also the longer key length improves security over WEP.

In this paper, we address this problem by using a novel cipher called Diffusioncipher. The proposed Diffusion cipher is similar in structure to AES with one important difference, in that, *the* Diffusion cipher can securely encrypt 'k' bit input data to 'n' bit encrypted output data.. This secure expansion property when used in the CTR mode results in higher encryption throughput. With the appropriate choice of parameters, one Diffusion cipher encryption in the CTR mode can encrypt, 288 bits of information as opposed to 128 bits using the AES. The Diffusion cipher requires only half the number of encryptions to encrypt the entire MPDU.

II. RELATED WORK

A. CCMP

Counter Cipher Mode with Block Chaining Message Authentication Code Protocol or CCMP (CCM mode Protocol) is an encryption protocol designed for Wireless LAN products that implement the standards of the IEEE 802.11i amendment to the original IEEE 802.11 standard. CCMP is an enhanced data cryptographic encapsulation mechanism designed for data confidentiality and based upon the Counter Mode with CBC-MAC (CCM) of the AES standard [17].

The CCMP is executed at the link layer of 802.11 wireless devices, where the input data called the message service data unit (MSDU) is often broken down into many message protocol data units (MPDUs), where each MPDU consists of MAC header and a data payload. Each of the MPDU's are processed using CCMP to produce encrypted MPDU's with tagged Message Integrity Check (MIC) value and clear text headers. The CCMP requires two state variables: a single shared key (*K*) for both encryption and authentication and a 48-bit packet sequence number (PN). The CCMP uses the PN to construct both the counter for encryption and the Initialization Vector (IV) for authentication. A 64-bit message integrity check (MIC) value is generated by encrypting the IV, the entire MPDU and the CCMP header with 128-bit AES in CBC- MAC mode, which is then appended to the MPDU. The data payload of the MPDU and the MIC are encrypted using the 128-bit AES in the CTR mode. The value of the counter is incremented after every 128-bit block encryption. At the receiver, the encrypted MPDU's are decrypted using the PN extracted from the clear text headers and the shared secret key.

CCMP uses CCM that combines CTR for data confidentiality and CBC-MAC for authentication and integrity. CCM protects the integrity of both the MPDU data field and selected portions of the IEEE 802.11 MPDU header. CCMP is based on AES processing and uses a 128-bit key and a 128-bit block size. A CCMP Medium Access Control Protocol Data Unit (MPDU) comprises five sections. The first is the MAC header which contains the destination and source address of the data packet. The second is the CCMP header which is composed of 8 octets and consists of the packet number(PN), the Ext IV, and the key ID. The packet number is a 48-bit number stored across 6 octets. The PN codes are the first two and last four octets of

the CCMP header and are incremented for each subsequent packet. Between the PN codes are a reserved octet and a Key ID octet. The Key ID octet contains the Ext IV (bit 5), Key ID (bits 6-7), and a reserved subfields (bits 0-4). CCMP uses these values to encrypt the data unit and the MIC. The third section is the data unit which is the data being sent in the packet. Lastly are the Message Integrity Code (MIC) which protects the integrity and authenticity of the packet and the frame check sequence(FCS) which is used for error detection and correction. For the above sections only the data unit and MIC are encrypted.

CCMP is the standard encryption protocol for use with the WPA2 standard and is much more secure than the WEP protocol and TKIP protocol of WPA. CCMP provides the following security services,

- Data Confidentiality; ensures only authorized parties can access the information.
- Authentication; provides proof of genuineness of the user.
- Access control in conjunction with layer management.

Because CCMP is a block cipher mode it is secure against attacks to the 2^{128} steps of operation if the key for the encryption is 256 bits or larger. Generic meet-in-the-middle attacks do exist and can be used to limit the theoretical strength of the key to $2^{(n/2)}$ (where n is the number of bits in the key) operations needed.

III. DESIGN OF DIFFUSION CIPHER

One way to improve the energy efficiency of the CCMP, is to introduce a cipher that can encipher larger chunks of the plaintext per encryption. That is, we need a high encryption throughput cipher that replaces the 128-bit AES used in the CCMP. We define the term *encryption throughput* of a cipher as the number of plaintext bits encrypted with one single encryption of the cipher. In this paper, we take a novel approach to increase the encryption throughput by introducing a Diffusionfunction that also causes expansion. This can be done by using a new class of codes that we call the Diffusioncode (Diffusion - codes) [11] at the Diffusionlayer. Although, these codes were designed with the specific goal of constructing error correcting ciphers [12], these can also be used to provide a means of securely increasing the encryption throughput.

A. *The Diffusion Cipher in GF(2^8)*

The DIFFUSIONcipher [13] is a key-alternating [5] block cipher, similar structure in AES. It composed of several iterations of the round transformation and key mixing operation. The round transformation consists of three oprations: a) the non- linear substitution layer, b) symbol transposition layer and c) the Diffusionencoding layer. The Diffusion encoding layer takes in '*k'* bits of input and produces '*n'* bits of output, where $k < n$ and at each round r. Note that, the Diffusion encoding is not performed in the final round. The key mixing layer follows every round transformation and is also performed once before the first round. The input data, as it goes through each round of the cipher, is referred to as the cipher state.

The round keys are generated using the key expansion algorithm, which is similar to that of the AES key expansion algorithm [8], to construct 11 round keys using initial key. Before proceeding the AES key expansion, we have to XOR operation with user key and Initialization Vector. It produces an initial key to round keys operation. All the operations in Diffusion cipher are performed in the finite field of order 2^8, denoted by GF (2^8). A detailed description of all the layers of Diffusion cipher [12] follows:

1) *Key Mixing Layer:* The key mixing layer is a simple bitwise XOR operation of the cipher state with the round key . For both encryption and decryption, only the key mixing layer stage makes use of the key. For this reason, the cipher begins and ends with an key mixing layer. For each round, a round key is derived from the main key using key generation layer, each round key is the same size as the state (36 bytes). The 128 bits of states are bitwise XORed with the 288 bits of the round key. This operation is viewed as a column wise operation between 6 bytes of a state column and one word(word consists of 6 bytes, that is one column) of a round key; it can also be viewed as a byte-level operation.

The output cipher state of the key mixing layer of round 'r-1' forms the input cipher state to the next round 'r'. The key size is 288 bits and it is represented by 6x6 matrix . The key generated is expanded to produce round key for each round.

The inverse key mixing layer transformation is identical to the key mixing layer transformation, because the XOR operation is its own inverse.

2) *Non-linear Substitution Layer:* The goal of the substitution step is to reduce the correlation between input and output bits. In substitution layer the input data bytes are substituted with the bytes in the S-Box .The non-linear byte substitution, operating on each byte of the state independently. This is a byte-by-byte substitution and the substitution byte for each input byte is found by using the S-Box. The S-Box is the 16x16 matrix which contains the hexadecimal byte values up to 255 in the random order. The data byte is splited as upper 4 bits and lower 4 bits . The byte in the intersection of the row ,column represented by the lower and upper bits is replaced in the place of the data byte. The following polynomial transformation to each unique bit of each byte in the S-box value is,

$$b_i' = b_i \oplus b_{(i+4) \bmod 8} \oplus b_{(i+5) \bmod 8} \oplus b_{(i+6) \bmod 8} \oplus b_{(i+7) \bmod 8} \oplus c_i \quad (1)$$

Where c_i is the i-th bit of byte c with the data value {e.g. 63: $(c_7c_6c_5c_4c_3c_2c_1c_0)$=(01100011)}, b_i is the S-box hexadecimal 8-bit value.

The S-Box values can either be calculated on-the-fly to save memory or the pre-calculated values can be stored in an array. The S-box is constructed by combining the inverse function with an invertible affine transformation. The S-box is also chosen to avoid any fixed points (and so is a derangement), and also any opposite

fixed points In decryption the cipher bytes are replaced with the bytes in the similar manner except that the inverse S-Box is used to reverse the process . The inverse S-box is constructed by applying the inverse of the transformation followed by taking the multiplicative inverse. The inverse transformation is:

$$b'_i = b_{(i+2) \bmod 8} \oplus b_{(i+5) \bmod 8} \oplus b_{(i+7) \bmod 8} \oplus d_i \quad (2)$$

3) *Symbol Transposition Layer:* The aim of this layer is to change the position of the data using permutation . This layer is used to reduce the linearity among the input data and making the attacks harder . The permutation function is stored in the array and it is used to make the transposition . The position of data bytes are changed using the permutation function. This transposition layer is more substantial and is treated as an array of 6-byte columns. Thus, a row shift moves an individual byte from one column to another, which is a linear distance of a multiple of 6 bytes. This transformation ensures that the 6 bytes of one column are spread out to 6 different columns. In decryption the reverse process is carried out i.e., the inverse permutation is applied to the cipher state . The same permutation function is used for the inverse permutation too . In matrix transposition transformation, any two symbols appearing in the same column before the transformation appear in different columns after the transformation. Hence, this transformation is a Diffusionoptimal transformation.

4) Diffusion *Encoding Layer:* The Diffusion encoding transformation is the second Diffusionoperations used in the Diffusion cipher. The aim of this layer is to diffuse the intra symbol avalanche caused by the substitution layer to a large number of symbols in the resulting cipher state. This layer is to securely expand the no. of bits. The Diffusion cipher can securely encrypt ‘ k ’ bit input data to ‘ n ’ bit encrypted output data (where k<n). This secure expansion property when used in the CTR mode results in higher encryption throughput. The Diffusion encoding transformation is the second Diffusionoperations used in the Diffusion cipher . In this layer, it replaces each byte of a column by a function of all the bytes in the same column. It operates on each column individually, each column of the input cipher state is multiplied with generator matrixes to obtain the output cipher state.

The six bytes of each column of the state are combined using an invertible linear transformation. The Diffusion layer function takes six bytes as input and output takes six bytes, where each input bytes affects all six output bytes. Each column is treated as a polynomial over GF(2^8) and is then multiplied modulo x^4+1 with a fixed polynomial $a(x)=x^5+3x^4+2x^3+x^2+x=1$. The coefficients are displayed in their hexadecimal equivalent of the binary representation of bit polynomial form. This multiplicative polynomial can be expanded and viewed as a circular matrix for entire row in the state matrix are followed as,

$$\begin{pmatrix} 1 & 1 & 3 & 2 & 1 & 1 \\ 1 & 1 & 1 & 3 & 2 & 1 \\ 1 & 1 & 1 & 1 & 3 & 2 \\ 2 & 1 & 1 & 1 & 1 & 3 \\ 3 & 2 & 1 & 1 & 1 & 1 \\ 1 & 3 & 2 & 1 & 1 & 1 \end{pmatrix} \quad (3)$$

In decryption the Inverse Diffusion layer process is carried out with the cipher state as input. Here, it performs multiplicative inverse polynomial modulo of x4+1 with a(x) =x5+3x4+2x3+x2+x+1 and the resulted circular matrix are,

b(x)=a-1(x) mod x4+1 (i.e.)

b(x) =9x5+0Ex4+09x3+0Dx2+11x+13 (4)

The inverse transformation matrix equal to the identity matrix. The coefficients in Inverse Diffusion layer are more formidable to implement.

IV SECURITY ANALYSIS OF DIFFUSIONCIPHERS

In this section, we analyze the security of Diffusion ciphers by looking at the resistance it offers against some well-known cryptanalytic attacks.

Differential cryptanalysis [4] is a chosen plaintext-cipher text attack that makes use of difference propagation property of a cipher to deduce the key bits. The difference propagation property of an S-box is the relative number of all input pairs that for the given input difference, give rise to a specific output difference. The difference propagation of consecutive round can be concatenated across several rounds to form a differential trail. The propagation ratio over all the rounds of a differential trail can be approximated by the product of the propagation ratios of its active S-boxes. Differential cryptanalysis can break the Diffusion cipher with complexity less than $O(2^{128})$ if the maximum possible propagation ratio over all rounds is significantly larger than 2^{i127}.

Linear cryptanalysis [13] is a known plaintext-cipher text attack that makes use of linearity in the cipher to obtain the key bits. The substitution is the only non-linear step in most of the block ciphers including the proposed Diffusion cipher. The linearity of an active S-box can be approximated to the maximum input-output correlation exhibited by it. The active S-boxes in a round are determined by the non-zero symbols in the selection vectors at the input of the round. The linearity of one round can be extended to multiple rounds to form a linear trail. The correlation (measure of linearity) of a linear trail (multiple rounds) can be approximated to the product of input-output correlations of its active S-boxes. Linear cryptanalysis can break the Diffusion cipher with complexity less than $O(2^{128})$ if the maximum possible correlation of any linear trail over all rounds is significantly larger than 2^{i64}.

Hence, a lower bound on the number of active symbols in any linear or differential trail will give a lower bound on the resistance of the cipher to linear and differential cryptanalysis. The S-boxes used in the substitution layer of Diffusion cipher have a maximum propagation ratio of 2^{-6} and maximum input and output correlation of 2^{-3}. This shows that there are no four round differential trails with predicted propagation ratio above 2^{-215} and no four round linear trails with predictable input output correlation above 2^{-105}. The initial six rounds are added as a security margin towards future attacks, just as in AES. Hence the 10 round 128 bit Diffusion cipher is secure against linear and differential cryptanalysis.

The Square attack [6] (also known as Integral attack or the Saturation attack) makes use of the byte oriented nature of the Square block cipher which was the predecessor of AES. As AES is also a byte oriented cipher, this attack has been extended to reduced versions of AES [9], [10]. The proposed Diffusion cipher also comprises of byte oriented operations which are loosely based on AES, hence Diffusion ciphers with fewer than seven rounds would be as weak as reduced versions of the AES. Although the Diffusion cipher is as secure as AES against most of the well-known attacks, the Diffusion cipher uses a larger key length to achieve the same security level as that of AES. Since, the key expansion is performed only once every session, its computational overhead is negligible.

IV. EXPERIMENTAL RESULTS

Two set of experiments were conducted, one on a laptop and one on the desktop. The test bed consists of a DELL Inspiron laptop with 2.27 GHz Intel i5 processor, 4GB RAM, running window 7 operating system and a desktop with 2.70 GHz Intel Pentium processor, 4GB Ram, running windows 7 operating system. The energy can be measured by executing both AES and Diffusion in the above specification.

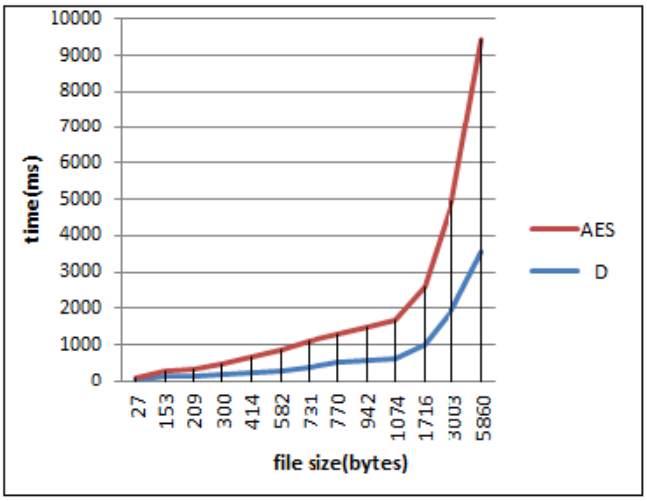

Fig. 1. The Time consumed per byte due to 128 bit AES and 288 bit Diffusion encryption on laptop and desktop.

Fig. 1 shows the time complexity comparison of AES and Diffusion. The running time taken from the encryption algorithms measured, as a function of starting time as well as ending time of execution. We measured the energy consumed by a full 10 round 288-bit (key brook length) Diffusion cipher, 10 round 128-bit AES on both desktop and laptop. Fig.5.1 plots the time (ms) per byte (file size in bytes) energy consumption due to AES and Diffusion cipher. .It can be observed from the Fig that, Diffusion cipher results 20% reduction in energy consumption, on both desktop and the laptop. We observe that Diffusion cipher consumes significantly less energy compared to the AES, as the frame length gets larger.

VI. CONCLUSION

In this paper, we proposed a Diffusion cipher security mechanism that securely encrypts 288-bits input states with larger 288-bits key stream. Replacing the AES with the Diffusion cipher allows us to achieve higher encryption throughput. Energy analysis experiments reveal that Diffusion cipher consumes less energy compared to the traditional AES. Also the proposed system performs significantly better when larger frame lengths are used, thereby demonstrating the significant energy gains that could be achieved in resource constrained systems.

In future the efficiency can be increased further by increasing frame size more than 288 bits and the implementation in parallel programing languages like open CL. This will reduces time complexity to the algorithm and also decrease the energy consumption during encryption.

REFERENCES

[1] C.N. Mathur and K.P.Subbalakshmi. “Energy Efficient Wireless Encryption”, Networking and Communications (MSyNC) Lab, 2005.

[2] Amendment 6: Medium access control (mac) security enhancements. *802.11i, Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications*, July 2004.

[3] "Wi-Fi Protected Access" (http:/ / www. wifialliance. org/knowledge_ center_overview. php? docid=4486). *Wi-Fi Alliance*. . Retrieved 2008-02-06

[4] E. Biham and A. Shamir. Differential cryptanalysis of the full 16-round des. In *CRYPTO ’92: Proceedings of the 12th Annual International Cryptology Conference on Advances in Cryptology*, pages 487–496, London, UK, 1993. Springer-Verlag.

[5] J. Daemen. *Cipher and hash function design strategies based on linear and differential cryptanalysis*. K.U.Leuven, March 1995.

[7] J. Daemen and V. Rijmen. *The design of Rijndael: AES — the Advanced Encryption Standard*. Springer-Verlag, 2002.

[8] H. Feistel. Cryptography and computer privacy. 228(5):15–23, May 1973.

[9] FIPS. Specification for the advanced encryption standard (AES). Federal Information Processing Standards Publication 197, 2001.

[10] H. Gilbert and M. Minier. A collision attack on 7 rounds of rijndael. In *AES Candidate Conference*, pages 230–241, 2000.

[11] S. Lucks. Attacking seven rounds of rijndael under 192-bit and 256-bit keys. In *AES Candidate Conference*, pages 215–229, 2000.

[12] C. N. Mathur, K. Narayan, and K. Subbalakshmi. Diffusioncodes: A class of maximum distance separable codes for error resilient block ciphers. *2nd IEEE International Workshop on Adaptive Wireless Networks (AWiN), Globecom*, November 2005.

[13] C. N. Mathur, K. Narayan, and K. Subbalakshmi. High Diffusioncipher: Encryption and error correction in a single cryptographic primitive. To appear in the 4th International Conference on Applied Cryptography and Network Security Conference (ACNS), June 2006.

[14] M. Matsui. Linear cryptanalysis method for DES cipher. In *Advances in cryptology -EUROCRYPT93, Lecture Notes in Computer Science*, volume 765, pages 1–11, 1993.

[15] K. Nyberg. Differentially uniform mappings for cryptography. In *EURO-CRYPT '93: Workshop on the theory and application of cryptographic techniques on Advances in cryptology*, pages 55–64, Secaucus, NJ, USA, 1994. Springer-Verlag New York, Inc.

[16] L. M. S. C. of the IEEE Computer Society. Wireless lan medium access control (mac) and physical layer (phy) specifications. 1999.

[17] J. Walker. 802.11 security series part ii: The temporal key integrity protocol (tkip). *Technical report, Platform Networking Group, Intel Corporation*.