AUSTRALIAN JOURNAL OF BASIC AND APPLIED SCIENCES

ISSN:1991-8178 EISSN: 2309-8414
Journal home page: www.ajbasweb.com

A Hybrid Cryptographic algorithm design using Block and Stream cipher based Confidentiality and Integrity in Wireless Sensors Networks

[1]M. Senthil Murugan and [2]T. Sasilatha

[1]*Research Scholar, Satyabhama University Associate Professor, Department of ECE St. Joseph's Institute of Technology Chennai, India*
[2]*Professor, Department of ECE Sree Sastha Institute of Engineering and Technology Chennai, India*

Address For Correspondence:
M. Senthil Murugan, Research Scholar, Satyabhama University Associate Professor, Department of ECE St. Joseph's Institute of Technology Chennai, India.
E-mail: senthilmuruganap@gmail.com

ARTICLE INFO

Article history:
Received 10 December 2015
Accepted 28 January 2016
Available online 10 February 2016

Keywords:
Wireless Sensor Network, Block Cipher, Stream Cipher, Advanced Encryption Standard, RC-4, Message Digest 5,

ABSTRACT

A wireless sensor network spatially distributed autonomous sensors to monitor physical or environmental conditions, such as temperature, sound, pressure, etc. A Group of sensor nodes deployed in particular environment constitute a wireless sensor network. For transmitting a data, image or video in a secured manner on wireless sensor network (WSN), cryptography plays an important role. To achieve security in wireless sensor network, it is important to be able to encrypt message send among sensor nodes. This paper present a hybrid model encryption algorithm which combines the features of block cipher and stream cipher techniques to achieve confidentiality and integrity. A hybrid encryption method enhances the security against attacks with minimized key maintenance.

INTRODUCTION

Sensor networks usually consist of a large number of ultra-small autonomous devices. Each device, called a sensor node, is battery powered and equipped with integrated sensors, for data processing and short-range radio communications. The application of wireless sensor network includes military sensing and tracking, environmental monitoring, patient monitoring and tracing, smart environments, etc. The majority of these application may be split into two classifications data collection and even detection. Where data collection is the goal, the sensors may be required to collect data for short periods of times in a day. In this case, most of the time the sensor node will be asleep for conserving power. However, whereas in event detection, such as detecting the ignition of a fire, it would be anticipated that the sensor nodes must remain awake, thus consuming their precious limited power. When sensor networks are deployed in a hostile environment, security becomes extremely important, as they may affect in different types of attacks such as Denial of Service attack, Sinkhole attack, Sybil attack, Wormhole attack and Hello Flood attack. To provide security, communication should be encrypted and authenticated.

In remainder of the paper is organized as follows: in section 2 security goals in sensor networks are discussed. In section 3 different types of attacks in wireless sensor networks are discussed. In section 4 different cryptography algorithm are discussed. In section 5 related work based on this paper are discussed. In section 6 the proposed methods are discussed and conclusion will be drawn.

Security Goals In Sensor Networks:

2.1 Data Confidentiality:

Confidentiality is the ability to conceal messages from a passive attacker so that any message communicated via the sensor network remains confidential (Dr. G. Padmavathi and Mrs. D. Shanmugapriya, 2009). This is the most important issue in network security. A sensor node should not reveal its data to the neighbors.

To Cite This Article: M. Senthil Murugan and T. Sasilatha., A Hybrid Cryptographic algorithm design using Block and Stream cipher based Confidentiality and Integrity in Wireless Sensors Networks. ***Aust. J. Basic & Appl. Sci.,*** *10(1): 387-393, 2016*

Australian Journal of Basic and Applied Sciences, 10(1) January 2016, Pages: 387-393

2.2 Data Integrity:

Data integrity in sensor networks is needed to ensure the reliability of the data and refers to the ability to confirm that a message has not been tampered, altered or changed. Even if the network has confidentiality measures, there is still a possibility that the data integrity has been compromised by alterations (Ian, F., 2002).

2.3 Data Authentication:

Authentication ensures the reliability of the message by identifying its origin. Attacks in sensor networks do not just involve the alteration of packets; adversaries can also inject additional false packets. Data authentication verifies the identity of the senders and receivers. Data authentication is achieved through symmetric or asymmetric mechanisms where sending and receiving nodes share secret keys. Due to the wireless nature of the media and the unattended nature of sensor networks, it is extremely challenging to ensure authentication.

3. Attacks In Wireless Sensor Networks:

Wireless sensor networks are vulnerable to security attacks due to the broadcast nature of the transmission medium. Basically attacks are classified as active attacks and passive attacks. The monitoring and listening of the communication channel by unauthorized nodes are known as passive attack (John Paul Walters, 2006). An unauthorized node monitors, listens and modifies the data stream in the communication channel are known as active attack, example of active attacks are Denial of Service attack, Sinkhole attack, Sybil attack, Wormhole attack and Hello Flood attack.

3.1 Denial of Service attack:

Denial of Service (DOS) is produced by the unintentional failure of nodes or malicious action. DOS attack is meant not only for the adversary's attempt to subvert, disrupt, or destroy a network, but also for any event that diminishes a network's capability to provide a service. In wireless sensor networks, several types of DOS attacks in different layers might be performed (Chris Karlof, David Wagner, 2003).

3.2 Sinkhole Attack:

Attracting traffic to a specific node in called sinkhole attack. In this attack, the adversary's goal is to attract nearly all the traffic from a particular area through a compromised node. Sinkhole attacks typically work by making a compromised node look especially attractive to surrounding nodes.

3.3 Sybil Attack:

A single node duplicates itself and presented in the multiple locations. The Sybil attack targets fault tolerant schemes such as distributed storage, multipath routing and topology maintenance. In a Sybil attack, a single node presents multiple identities to other nodes in the network. Authentication and encryption techniques can prevent an outsider to launch a Sybil attack on the sensor network.

3.4 Wormhole Attack:

In the wormhole attack, an attacker records packets (or bits) at one location in the network, tunnels them to another location, and retransmits them into the network.

3.5 Hello Flood Attack:

An attacker sends or replays a routing protocol's HELLO packets from one node to another with more energy. This attack uses HELLO packets as a weapon to convince the sensors in WSN. In this type of attack an attacker with a high radio transmission range and processing power sends HELLO packets to a number of sensor nodes that are isolated in a large area within a WSN. The sensors are thus influenced that the adversary is their neighbor. As a result, while sending the information to the base station, the victim nodes try to go through the attacker as they know that it is their neighbor and are ultimately spoofed by the attacker.

4. Cryptographic Algorithms:

Cryptographic algorithms can be classified as Symmetric and Asymmetric, in symmetric cryptographic algorithms same key are used for encryption and decryption and in asymmetric cryptographic algorithm two keys are used for encryption and decryption.

In symmetric cryptographic techniques as shown in Fig 2, a single shared key is used between the two communicating nodes for encryption and decryption. This key has to be kept secret in the network, which can be quite hard in the exposed environment where WSNs are used. Most security schemes for WSN uses only symmetric cryptography, due to its ease of implementation on limited hardware and small energy demands, especially if the implementation is done in hardware to minimize performance loss. Two types of symmetric ciphers encryption are block ciphers and stream ciphers. The block cipher approach work on blocks of a specific fixed length. The stream ciphers are bitwise encryption on the data.

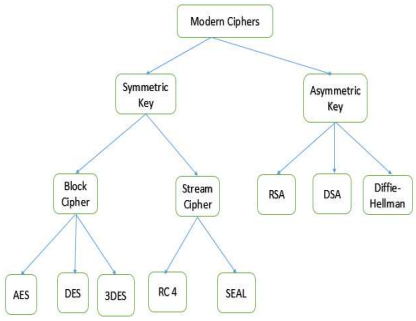


Fig.1: Modern Ciphers.

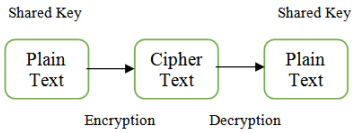


Fig. 2: Symmetric Key.

4.1 ***Block Cipher:***

In cryptography, a block cipher is a deterministic algorithm operating on fixed-length group of bits, called *blocks*, with an unvarying transformation that is specified by a symmetric key. Block ciphers are important elementary components in the design of many cryptographic protocols, and are widely used to implement encryption of bulk data. Product ciphers were suggested and analyzed by single author Claude Shannon in his seminal 1949 publication Communication Theory of Secrecy Systems as a means to effectively improve security by combining simple operations such as substitution and permutations. Iterated product ciphers carry out encryption in multiple rounds, each of which uses a different sub key derived from the master key. One widespread implementation of such ciphers is called a Feistel network, named after Horst Feistel, and notably implemented in the DES cipher. Many other realizations of block ciphers, such as the AES, are classified as substitution-permutation networks. Although military algorithms are traditionally kept secret, this is infeasible for open commercial use, and everyone in the field knows the AES algorithm. The AES block size is 128 bits (before the AES, 64 bits was common) and the key sizes can be either 128, 192, or 256 bits. Every key defines a different codebook, mapping each plaintext value to a unique cipher text value.

4.2 Aes Algorithm:

AES is the new encryption standard recommended by NIST to replace DES in 2001. AES algorithm can support any combination of data (128 bits) and key length of 128, 192, and 256 bits. The algorithm is referred to as AES-128, AES-192, or AES-256, depending on the key length. During encryption-decryption process, AES system goes through 10 rounds for I28-bit keys, 12 rounds for I92-bit keys, and 14 rounds for 256-bit keys in order to deliver final cipher-text or to retrieve the original plain-text. AES allows a 128 bit data length that can be divided into four basic operational blocks. These blocks are treated as array of bytes and organized as a matrix of the order of 4×4 that is called the state. For both encryption and decryption, the cipher begins with an Add Round Key stage. However, before reaching the final round, this output goes through nine main rounds, during each of those rounds four transformations are performed; 1) Sub-bytes, 2) Shift-rows, 3) Mix-columns, 4) Add round Key. In the final (10th) round, there is no Mix-column transformation. Decryption is the reverse process of encryption and using inverse functions: Inverse Substitute Bytes, Inverse Shift Rows and Inverse Mix Columns.

Each round of AES is governed by the following transformations

4.2.1 Substitute Byte transformation:

AES contains 128 bit data block, which means each of the data blocks has 16 bytes. In sub-byte transformation, each byte (8-bit) of a data block is transformed into another block using an 8-bit substitution box which is known as Rijndael Sbox.

4.2.2 ***Shift Rows transformation:***

It is a simple byte transposition, the bytes in the last three rows of the state, depending upon the row location, are cyclically shifted. For 2nd row, 1 byte circular left shift is performed. For the 3rd and 4th row 2-byte and 3-byte left circular left shifts are performed respectively.

4.2.3 Mixed columns transformation:

This round is equivalent to a matrix multiplication of each Column of the states. A fix matrix is multiplied to each column vector. In this operation the bytes are taken as polynomials rather than numbers.

4.2.4 Add round key transformation:

It is a bitwise XOR between the 128 bits of present state and 128 bits of the round key. This transformation is its own inverse.

4.3 Straem Cipher:

A stream cipher is a symmetric key cipher where plaintext digits are combined with pseudorandom cipher digit stream (key stream). In a stream cipher each plaintext digit is encrypted one at a time with the corresponding digit of the key stream, to give a digit of the cipher text stream. Since encryption of each digit is dependent on the current state of the cipher, it is also known as *state cipher*. In practice, a digit is typically a bit and the combining operation an exclusive-OR. The pseudorandom key stream is typically generated serially from a random seed value using digital shift registers. The seed value serves as the cryptographic key for decrypting the cipher text stream.

Stream ciphers are often used for their speed and simplicity of implementation in hardware, and in applications where plaintext comes in quantities of unknowable length like a secure wireless connection. If a block cipher (not operating in a stream cipher mode) were to be used in this type of application, the designer would need to choose either transmission efficiency or implementation complexity, since block ciphers cannot directly work on blocks shorter than their block size.

Another advantage of stream ciphers in military cryptography is that the cipher stream can be generated in a separate box that is subject to strict security measures and fed to other devices such as a radio set, which will perform the XOR operation as part of their function.

RC4 is the most widely used stream cipher

4.4. RC4 Algorithm:

RC4 is a binary additive stream cipher. It uses a variable sized key that can range between 8 and 2048 bits in multiples of 8 bits (1 byte). This means that the core of the algorithm consists of a key stream generator function. This function generates a sequence of bits that are then combined with the plaintext with XOR. Decryption consists of re-generating this key stream and XOR'ing it to the cipher text, undoing it. The other major part of the algorithm is the initialization function in which accepts a key of variable size and uses it to create the initial state of the key stream generator. This is also known as the key schedule algorithm. RC4 is actual a class of algorithms parameterized on the size of its block. This parameter, n, is the word size for the algorithm. This is recommended n = 8, but for analysis purposes it can be convenient to reduce this. Also, for extra security it is possible to increase this value. The internal state of RC4 consists of a table of size 2n words and two word sized counters. The table is known as the S-box, and will be known as S. It always contains a permutation of the possible 2n values of a word. The two counters are known as i and j.

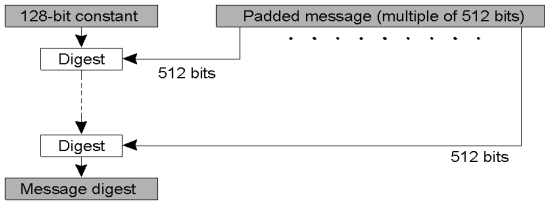


Fig. 3: Structure of MD 5 Algorithm.

The Key Schedule Algorithm of RC4. It accepts as input the key stored in K, and is l bytes long. It starts with the identity permutation in S and, using the key, continually swapping value to produce a new unknown key-dependent permutation. Since the only action on S is to swap two value, the fact that S contains a permutation is always maintained.

4.4.1 RC4 Key Schedule Algorithm:

Initialization: For i = 0 to 2n − 1
S[i] = i
j = 0
Scrambling: For i = 0 to 2n – 1

j = j + S[i] + K[i mod l]
Swap(S[i], S[j])

4.4.2 RC4 Pseudo Random Generation Algorithm:

The RC4 key stream generator works by continually shuffling the permutation stored in S as time goes on, each time picking a different value from the S permutation as output. One round of RC4 outputs an n bit word as key stream, which can then be XOR with the plaintext to produce the cipher text.

Initialization:
i = 0
j = 0
Generation Loop:
i = i + 1
j = j + S[i]
Swap(S[i], S[j])
Output z = S[S[i] + S[j]]

4.5 Message Digest 5:

MD5 algorithm was developed by Professor Ronald L. Rivest in 1991. According to RFC 1321, "MD5 message-digest algorithm takes as input a message of arbitrary length and produces as output a 128-bit "fingerprint" or "message digest" of the input. It is conjectured that it is computationally infeasible to produce two messages having the same message digest, or to produce any message having a given pre specified target message digest. The MD5 algorithm is intended for digital signature applications, where a large file must be "compressed" in a secure manner before being encrypted with a private (secret) key under a public-key cryptosystem such as RSA." MD5 is considered one of the most efficient algorithms currently available and being used widely today.

4.5.1 The structure of MD5 algorithm:

MD5 algorithm uses four rounds, each applying one of four non-linear functions to each sixteen 32-bit segments of a 512-bit block source text. The result is a 128-bit digest. Figure 3 is a graph representation that illustrates the structure of the MD5 algorithm.

5. Related Work:

5.1 Subasree Security Protocol:

The plain text is encrypted with the help of ECC and the derived cipher text is communicated to the destination through any secured channel. Simultaneously, the Hash value is calculated through MD5 for the same plain text, which already has been converted into the cipher text by ECC. This Hash value has been encrypted with DUAL RSA and the encrypted message of this Hash value also sent to the destination. In this protocol, it is difficult to extract the plain text from the cipher text, because the Hash value is encrypted with DUAL RSA and the plain text is encrypted with ECC. The Hash value is calculated with MD5. However, there are two disadvantages. First, the message is encrypted by Asymmetric Algorithms (ECC and DUAL RSA Public key encryptions) that are slow compared to symmetric encryption. Second, if an attacker determines a person's private key, his or her entire messages can be read (Subasree, S. and N.K. Sakthivel, 2010).

5.2 Dubal Security Protocol:

The given plain text is encrypted with the help of key that is generated by ECDH. The encryption algorithm used is DUAL RSA, which takes as the original information and the key. The derived cipher text is appended with the digital signature for more authentications, generated by the ECDSA algorithm. Simultaneously, the Hash value of this encrypted cipher text is taken through the MD5 algorithm. Then, the generated cipher text and the signature is communicated to the destination through any secured channel. On the other side, i.e., on decryption end, the Hash value is first evaluated and integrated. Thereafter, the decryption of cipher text is done by DUAL RSA. Hence, the plaintext can be derived. In this protocol, the intruder may be trapped by both the encryption by the DUAL RSA with the key generated by ECDH algorithm and the appended signature. However, the used Asymmetric Encryption Algorithms (DUAL RSA and ECDH) are slow compared to symmetric encryption. In addition, the attacker may read the messages if he/she can determine the private key (Dubal, M.J., 2011).

5.3 Kumar Security Protocol:

The given plain text is encrypted first with AES algorithm and then with ECC algorithm. The Hash value of this encrypted cipher text is taken through the MD5 algorithm. On the other side, the Hash value is first evaluated and integrated. Thereafter, the decryption of cipher text is done by AES and ECC decryption

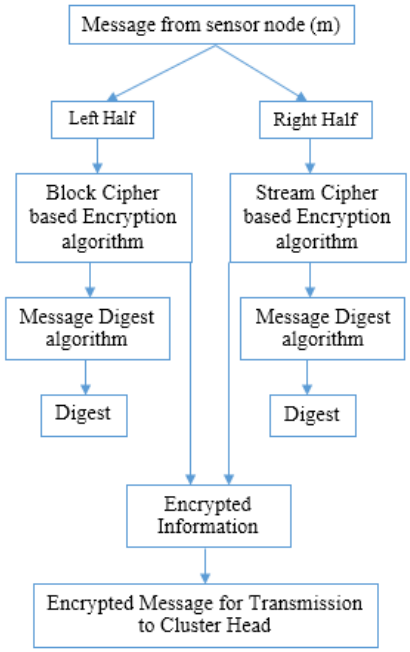
algorithms. Hence, the plaintext can be derived. This Protocol is a combination of both the Symmetric and Asymmetric Cryptographic Techniques. However, the execution time of this protocol is long because the plaintext is encrypted sequentially by both AES and ECC (Kumar, N., 2012)

5.4 Zhu protocol:

The plaintext is encrypted with Symmetric cipher algorithm, and the key and digital signature belonged to the Symmetric encryption algorithm are encrypted with Asymmetric key algorithm. The sender encrypts the plaintext P with the key KAES belonged to the AES algorithm. To ensure the security of the cipher algorithm and simplify the key management, the sender uses the key KAES only once. The receiver obtains the original information P after signature verification. This protocol suffers from low security level since that the message is encrypted in a single phase which leads to less complexity (Zhu, S., 2011).

Model Of Proposed Hybrid Encryption:

Algorithm:



Step 1: Initially the cluster head receives the message from sensor nodes.
Step 2: The Entire message will be divided as two halves left half and right half equally by size.
Step 3: The Left half of the message will be encrypted by using block cipher approach.
Step 4: The Right half of the message will be encrypted by using stream cipher approach.
Step 5: Form the encrypted information (Cipher text), the message digests are calculated in both the sides
Step 6: Finally the encrypted message and Digest values are concatenated and transmitted to next cluster head.

Conclusion:

In this paper, we have proposed a hybrid model encryption algorithm for wireless sensor networks. The proposed scheme is based on Block cipher and stream cipher symmetric key. This method tries to trap the eves by splitting the plain text and then applied two different symmetric key algorithm. First it takes the features of Block cipher second it takes the features of Stream cipher. In addition, Hashing is also used for Data integrity.

REFERENCE

Dr. G. Padmavathi and Mrs. D. Shanmugapriya, 2009. “A Survey of Attacks, Security Mechanisms and Challenges in Wireless Sensor Networks” IJCSIS, 4: 1-2.

Ian, F., Akykildiz, Weilian Su, Yogesh Sankarasubramaniam, Erdal Cayirci, 2002. “A Survey on Sensor Networks”, IEEE Communication Magazine.

John Paul Walters, Zhengqiang Liang, Weisong Shi, Vipin Chaudhary, 2006. “Wireless Sensor Network security: A Survey”, Security in Distributed, Grid and Pervasive Computing Yang Xiao, 3-5: 10-15.

Chris Karlof, David Wagner, 2003. “Secure Routing in Wireless Sensor Networks: Attacks and Countermeasures”, AdHoc Networks (Elsevier), 299-302.

Subasree, S. and N.K. Sakthivel, 2010. "Design of a new security protocol using hybrid cryptography algorithms," IJRRAS, 2(2): 95-103.

Dubal, M.J., T.R. Mahesh, P.A. Ghosh, 2011. "Design of a new security protocol using hybrid cryptography architecture," In Proceedings of 3rd International Conference on Electronics Computer Technology (ICECT), 5.

Kumar, N., 2012. "A Secure communication wireless sensor networks through hybrid (AES+ECC) algorithm", LAP LAMBERT Academic Publishing, 386.

Zhu, S., 2011." Research of hybrid cipher algorithm application to hydraulic information transmission," In Proceedings of International Conference on Electronics, Communications and Control (ICECC).