Security Issues and Solutions in Wireless Sensor Networks

Chérif Diallo *, Abdoulaye Sawaré, Maimouna Tedy Sow

Laboratoire Algèbre, Cryptographie, Codes et Applications (ACCA),
UFR Sciences appliquées et de Technologies (UFR SAT),
Université Gaston Berger BP 234 Saint-Louis, Sénégal
* cherif.diallo@ugb.edu.sn

***Abstract*—In a wireless sensor network (WSN), sensors are typically deployed in an area in order to collect events related to one or more sensing features such as temperature, humidity, light, pressure, magnetic, accelerometer, acoustic activity, seismic movements, etc. The sensors communicate with each other to relay events to a Sink node or a Base Station that communicates with the network operator. Since the most critical feature in sensor networks is the hardship of its energy resources because each sensor node has initial low resources in terms of energy and it is not often possible to replace batteries during network operations. Moreover, equipped with a small processor generally associated to a small storage unit, sensors often operate without human assistance, and could be deployed in open and hostile environments. Therefore, wireless sensor networks are subject to various types of threats, vulnerabilities and attacks which are carried out in order to disclosure sensitive information, to disrupt network operations and to exhaust the batteries of the nodes and then to reduce their lifetimes, and consequently that of the network. In this paper, we will try to give an overview study of security issues and solutions in WSN.**

I. Introduction

With the parallel evolution of micro-technology, embedded systems and the field of ubiquitous computing, many applications continue to be designed in the field of wireless sensor networks (WSNs). WSNs are ad hoc networks in which the nodes are sensors that can be considered as small chips [Fig. 1].

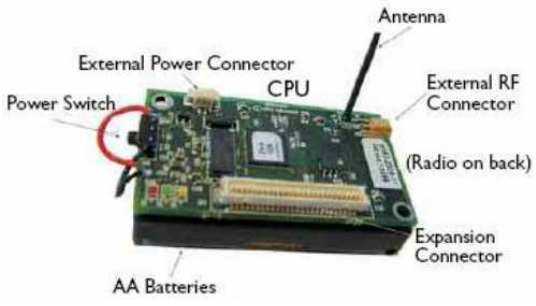


Fig. 1. Small chip sensor node

A sensor node [Fig. 1] is mainly composed by four basic units [Fig. 2]: the sensing unit, the processing unit, the wireless transceiver unit, and the power unit. It can also contain, depending on its domains of application, supplementary modules such as a Global Positioning System (GPS) or an energy generator system (solar cell or photovoltaic). There are also sensors, more voluminous, equipped with a mobility system responsible for moving the node in case of need.

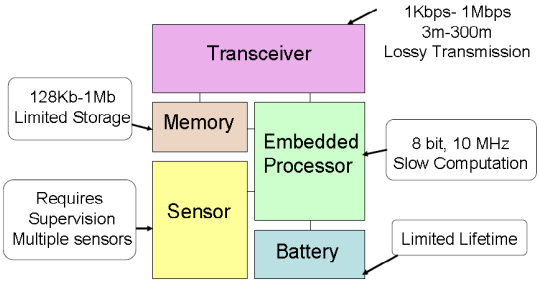


Fig. 2. Main basic units of a sensor node.

A Wireless Sensor Network (WSN) [Fig. 3] consists of a set of nodes deployed in a given environment (target) for accurate application (temperature, pressure, control, monitoring, intrusion detection, air humidity, agriculture, home automation, medical field, etc.). The sensors communicate with each other to relay information to a Sink node that communicates with the user interface.

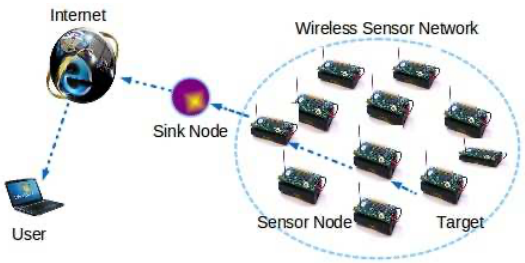


Fig. 3. Illustration of a wireless sensor network [1]

In a WSN, depending on the nature of the application, the number of nodes can be very large (several hundreds or thousands of nodes) and the deployment area is often hostile.

https://sites.google.com/site/ijcsis/
ISSN 1947-5500

For instance, in many military applications, sensors are deployed to monitor a battlefield or an enemy zone. Thereby WSNs are particularly vulnerable to many threats according to the nature of the deployment conditions (hostile area, open environment, no human presence, etc.), the low capacity of the sensors (limited storage, slow computation, limited lifetime battery) and the type of sensor communication channel (radio wave, lossy transmission). Furthermore, the sensors could suffer from physical attacks leading to the retrieval of confidential information contained in their memory such as cryptographic keys, certificates, etc.

On the other hand, the data collected by the sensor nodes should be sent to a sink node or a base station which may be a particular node or a client connected to the Internet network. Thus malicious nodes as well as attackers can access and manipulate data. Moreover, an attacker may attempt to add data into the network. Therefore, suitable confidentiality, integrity, authentication, availability and data freshness methods become essential to protect WSN against threats [2].

In this paper, we will focus on security issues and solutions in WSN. It is organized as follows. Firstly, in section II, we will talk about security requirements before presenting a short overview of attacks on WSNs in section III. In section IV we will present general security primitives that can be used to address security issues. Then, we will give a brief overview of authentication and control access schemes in section V. Finally, in section VI, we will present some principles of securing network operations.

II. WSN Security requirements

A. Energy saving

Given the limited energy of the sensors and deployment conditions without human presence which often exclude the possibility of replacing batteries, energy saving is a fundamental requirement for wireless sensor networks security. Indeed, disturbing the operation of a WSN could consist in simply setting up mechanisms contributing to exhaust the batteries of the nodes so as to reduce their lifetime as well as that of the network. Thus, many attacks are intended to exploit this vulnerability. Therefore, energy conservation must be taken into account in any implementation process of solutions for sensor applications.

B. Data confidentiality

Data confidentiality deals with mechanisms which ensure that information is only accessible to those authorized to access. This means that the information transmitted by each sensor node is only readable by the nodes to which that information is addressed. For sensor networks, data confidentiality is vital for most military, medical, industrial, and other applications.

C. Data integrity

Data integrity refers to the state of data which, during processing, storage or transmission, is not subjected to any deliberate or accidental alteration or destruction. Data integrity consists of four elements: completeness, accuracy, authenticity and validity. Then data integrity is one of the basic security criteria in information technology. It ensures that data is not altered by an unauthorized user [3]. Since the collected data should be sent to an operator, any alteration leads to misinterpretation, hence the need for suitable WSN integrity controls should be addressed in order to enhance accuracy of data transmission and processing of important network protocols such as routing, clustering, etc.

D. Availability

Availability is a security criterion which means that the system must operate flawlessly during the expected ranges of use and guarantee access to the services and resources with the expected response time. The availability of an equipment or a system could be translated as a performance measure that is obtained by dividing the length of time that the equipment or system is operational by the total time it would have been desired to be. In WSN, It ensures that the network is ready to operate properly and that the data are available. Then WSN is threatened by malicious nodes, intruders, attackers, etc, which could alter availability. Thereby, it is highly recommended to set up security mechanisms to prevent this risk [4].

E. Authentication

Authentication refers to the process of confirming that a principal is the one he claims to be. It concerns both sensor nodes and data. Mutual authentication of nodes may help in reducing malicious nodes to operate inside the network. The use of authentication techniques also allows a recipient to verify that received messages come from a reliable source node. While the collected data must be sent to an end user, it is then necessary to set up authentication policies to ensure the authenticity of data.

F. Authorization

Authorization is the function that specifies access rights to resources. Authentication and authorization should not be confused because they are two different processes. After authentication, the authorization determines the access rights of the authenticated system to the accessed objects. Then, the authorization techniques ensure that only authorized entities communicates across the network. In the WSN infrastructure, only communications between legitimate entities (nodes and users) must be permitted. It is important to ensure that network access is only possible for the authorized nodes, and also for the users which only have the required rights. Consequently, authorization mechanisms are also intended to block unauthorized access to the network.

https://sites.google.com/site/ijcsis/
ISSN 1947-5500

G. Data freshness

In sensor networks, the transmitted data are often derived from regularly repeated measurements. Therefore, it is essential to be able to ensure that information is recent or not in order to guarantee the reliability of WSN applications. Then data freshness implies that the data is recent, and it ensures that an adversary has not replayed old messages. The data freshness is a security criterion which helps to fight against replay attacks.

III. Attacks on WSNs

In this section we describe a non-exhaustive but representative list of the most common and known attacks in WSNs, which can be classified in two main categories: passive or active attacks. A passive attack is one in which the intruder eavesdrops but does not modify the message stream in any way. An active attack is one in which the intruder may transmit messages, replay old messages, modify messages in transit, or delete selected messages from the wire [5].

A. Passive listening or eavesdropping

This attack consists of listening without altering the data or the operation of the network. It is generally undetectable but prevention is possible. Also called eavesdropping, it is a network layer attack that focuses only on capturing packets transmitted from the network by other nodes and reading the data content in search of any type of information. This type of network attack is generally one of the most effective as a lack of encryption services is used.

B. Traffic analysis

The attacker analyzes the paths taken by the packets to recover valuable information about the vulnerabilities of the network. Traffic analysis can allow an attacker to know the location of base stations or data aggregation nodes by locating locations where the greatest number of packets pass through.

C. Man-in-the-Middle Attack

A Man-in-the-Middle attack [6][7] is an attack in which the attacker intercepts information ex-changed between two nodes. The two legitimate nodes think they are communicating directly with each other and do not suspect the presence of the intruder. This one could create and control a communication channel to eavesdrop, analyze traffic, insert, and block arbitrary messages within it.

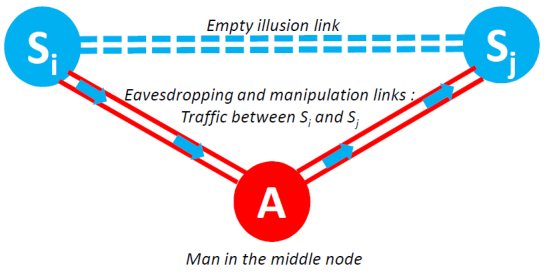


Fig. 4. Illustration of a Man-in-the-Middle Attack

The attacking node (A) relays messages between two legitimate nodes (S_i and S_j), creating the illusion that S_i and S_j are connected [Fig. 4]. The attacker (A) receives messages from both S_i and S_j and then creates a fake communication link between S_i and S_j. Both S_i and S_j are convinced that they are communicating directly without any intermediate node, whereas in reality this is not the case. The Man-in-the-Middle attack is a common threat during key exchange [6][7].

D. Hello flood attack

With a high power transmitter, an attacker performs a hello flood attack by broadcasting Hello messages on reaching certain network nodes which in turn will try to join the striker who is actually out of range [Fig. 5]. A repetition of such an attack would reduce energy nodes that will be forced to unnecessarily respond to every request of the attacker.

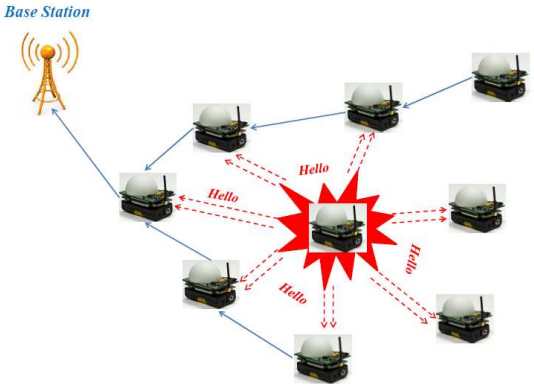


Fig. 5. Illustration of a Hello Flood Attack

E. Flooding

In this type of attack, an attacker uses one or more malicious nodes or a particular device which, in some cases, has strong transmission power, to regularly send messages on the network in order to saturate it. This is an active attack that is of the same type as denial-of-service attacks in conventional networks.

F. Replay attack

In a replay attack, an attacker intercepts the packets of a valid session and then replays they later [Fig. 6]. A replay attack is a form of network attack in which a valid data transmission is maliciously or fraudulently repeated or delayed [8].

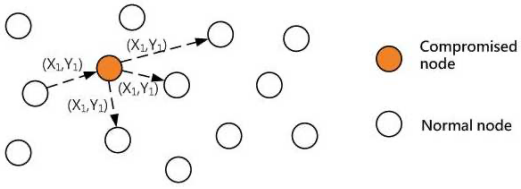


Fig. 6. Illustration of a Replay Attack

https://sites.google.com/site/ijcsis/
ISSN 1947-5500

To illustrate this attack, one can consider as an example a sensor network designed for fire detection purpose: if a first case of fire is detected and then a sensor node sends an alert packet to inform the control center, the attacker can register this packet, even if it is encrypted, and then retransmits it later. This will cause a new fire detection alert different from the first one. This attack is possible if the packet does not contain information concerning the sending date and time or if this date is accessible and easily modifiable by an attacker.

G. Message insertion

The attacker seeks in various ways (using malicious nodes, sending packets on the same radio frequency, etc.) to insert messages into the network traffic. This attack makes it possible to saturate, disrupt the network, or to deceive it by sending false information.

H. Message alteration

A malicious node will retrieve a message and alter it, adding false information (about the recipient, the sender, the information itself) or modifying it.

I. Byzantine attack

The Byzantine attack is described in [9] where the authors consider a WSN designed for the task of distributed binary detection. Their network consists of n sensors, each making an independent and identically distributed observation about the State of the Nature (said H_1 or H_0). These observations are successively delivered to a common fusion center for the final decision on the underlying statistical hypothesis. In a Byzantine attack [Fig. 7], malicious nodes which are referred to as the Byzantines sensors [10] will cooperatively works against the network by delivering data according to certain fictitious distributions properly designed in order to impair the detection capability of the fusion center [9].

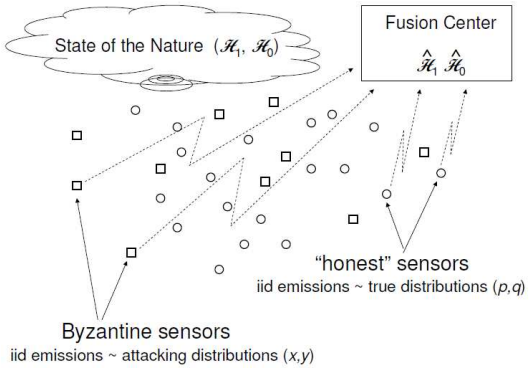


Fig. 7. Illustration of a Byzantine Attack [9]

J. Wormhole attack

The wormhole attack [Fig. 8] is an attack in which an attacker intercepts packets in one location and forwards them to another location of the network. With this attack in the network, non-neighboring nodes may believe they are neighbors, which somewhat disturbing routing and clustering operations.

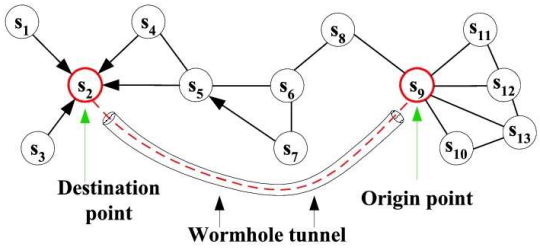


Fig. 8. Illustration of a Wormhole Attack

K. Sybil attack

In a Sybil attack [11], a malicious node holds simultaneously multiples identities in the network [Fig. 9]. This type of attack can affect the clustering operations by choosing a bad cluster head or the aggregation by distorting aggregation results.

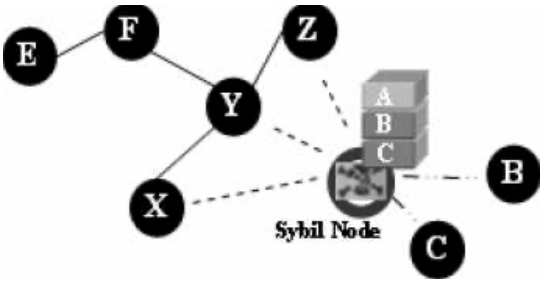


Fig. 9. Illustration of a Sybil Attack [11]

L. Sinkhole attack

In a sinkhole attack, an attacker compromises a node and manages to make it attractive against other nodes [Fig. 10]. With this attack, the normal nodes can for each operation, send data to the malicious node. The sinkhole attack is a particularly severe attack which prevents the base station from obtaining complete and correct sensing data [8].

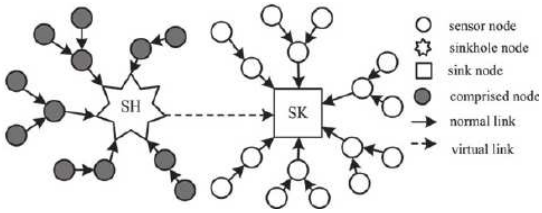


Fig. 10. Illustration of a Sinkhole Attack

M. Black hole attack

In wireless sensor networking, black hole attack refers to introduce malicious nodes in the locations where incoming or outgoing traffic could be silently discarded or dropped, without informing the source that the data did not reach its intended recipient. In a black hole attack, the black hole nodes themselves are often invisible [Fig. 11], and could be detected by monitoring the lost traffic.

https://sites.google.com/site/ijcsis/
ISSN 1947-5500

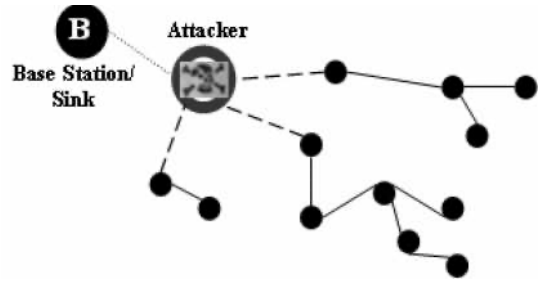


Fig. 11. Illustration of a Black hole attack

N. Gray hole or selective forwarding attack

In this type of attack, an attacker can compromise a node which could choose to forward messages or not. In the gray hole attack the malicious node could act as a normal node and then choose to drop some messages which are passing through it, while transmitting other packets. In this case, important messages might not reach their final destinations. This type of attack can disrupt the normal operation of applications and cause a possible denial of service (DoS).

O. Node Capture Attack

Most WSNs are often deployed in hostile areas that do not allow human monitoring of all sensors. It is then quite possible for an attacker to compromise a sensor. This node controlled by the attacker will allow him to access the network to retrieve sensitive information or to launch other attacks from this node. Thus, in Node Capture Attack, an attacker physically takes control of a sensor node and then compromises it so that stored data become inaccurate or manipulated. Capturing a node enables an adversary not only to get a hold of cryptographic keys and protocol states, but also to clone and redeploy malicious nodes in the network [8].

P. Sensor type specific attack

In this type of attack, an attacker modifies the physical behavior of the sensor. For instance, he can light a flame in front of a thermal sensor or light a lamp in front of a light sensor to deceive sensing operation. For this purpose, the compromised node will constantly send alert or continue recording false information until its energy is exhausted.

Q. Attacks on pacemakers

More and more medical implants have wireless interfaces to configure, store and retrieve information or remotely monitor patients. Cardiac implants (defibrillators, pacemakers) are good examples. Indeed, the pacemaker periodically emits small electrical stimuli to the heart and the defibrillator emits more powerful stimuli to restore a normal rhythm. Current systems provide no security (access control, integrity, privacy). Thus, an attacker can easily read the measured and stored data, read patient data (name, physician), modify the parameters or even disable the implant. So the attack on the pacemakers is a perfect example of specific attack that could have a dramatic effect on the patient's health. This shows the dangerousness of a lack of security control for medical sensors.

R. Jamming attack

In a jamming attack [Fig. 12], an attacker sends a powerful radio signal that will interfere with the sensor signals in order to disrupt legitimate communications and then cause a DoS attack.

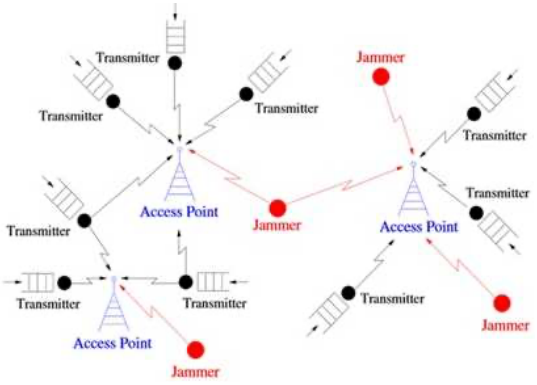


Fig. 12. Illustration of a Jamming Attack

	Passive Attacks	Active Attacks
Data	Passive Listening Man-in-the-Middle (eavesdropping) Traffic analysis	Hello flood, Flooding Replay, Message insertion, alteration Man-in-the-Middle (manipulation) Byzantine Wormhole, Sinkhole, Black hole Gray hole or selective forwarding
Sensors		Sybil Sinkhole Black hole Gray hole or selective forwarding Node Capture Sensor type specific Attacks on pacemakers
Signal		Jamming attack

TABLE I
PASSIVE AND ACTIVE ATTACKS

The figure [Fig. 13] lists WSN attacks related to each security requirements.

There are many other possible attacks in sensor networks, including attacks related to the top layer of TCP/IP applications such as repudiation, data corruption, hijacking session, SYN flooding, location disclosure attacks, DoS, impersonation, cryptographic primitive attacks, etc. As we can see, WSN security remains a real challenge sometimes due to the nature of deployment and sensors constraints. Several security techniques are proposed, but it should be noted that some of them are inappropriate. On the other hand, others seem well suited to sensors. In the next section, we will see some security primitives especially cryptography based mechanisms.

IV. Security primitives

As the number and type of attacks continue to increase, several security mechanisms have been designed to address this critical issue. In the following we present some of the most common security solutions.

https://sites.google.com/site/ijcsis/
ISSN 1947-5500

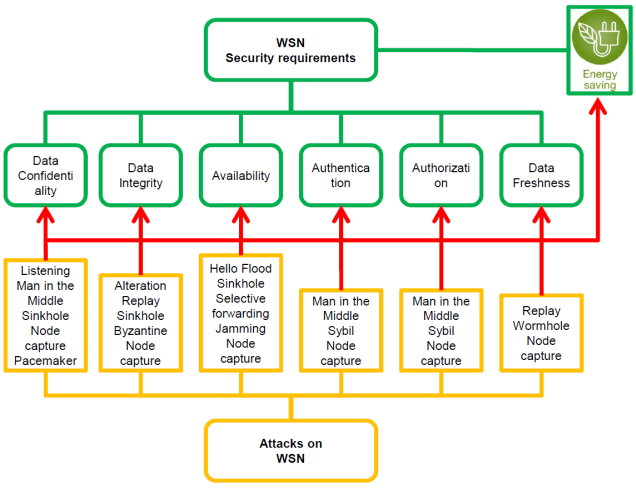


Fig. 13. Attacks on WSNs

A. Symmetric Key Encryption Primitives

There are basically two types of symmetric key encryption primitives: block ciphers and stream ciphers [12]. The block cipher consists in dividing the data packet into small sizes. In WSN, this will help in managing energy consumption and ensure data security. For the stream cipher, instead of dividing data into small blocks, the algorithm treats them in a linear way, bit by bit, and takes into account of any data length.

The Advanced Encryption Standard (AES) algorithm is part of the block cipher encryption family. It uses fixed 128 bits blocks for providing keys of 128, 192 and 256 bits length. This algorithm consists of a sequence of substitutions and permutations. It uses a 4x4 matrix initialized with the value of the encryption code which is changed at every round. For encryption, each round corresponds to a sequence of four steps [12]: AddRoundKey, where the subkey is combined with the state using the XOR operation, SubBytes, where each byte in the state is replaced with its entry in a fixed 8 bits lookup table, ShiftRows, where bytes in each row of the state are shifted cyclically to the left, and MixColumns, where each column of the state is multiplied with a fixed polynomial $c(x)$.

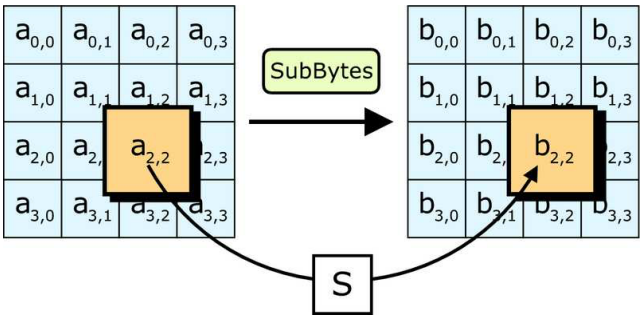


Fig. 14. AES SubBytes step

The PRESENT [13] algorithm is an ultra-lightweight block cipher encryption which works on 64 bits blocks and produces keys of 80 and 128 bits length. For better implementation, the authors advocate the use of 80 bits key length. This algorithm takes place on 31 rounds. For each round, it performs an XOR operation to introduce an intermediate key $K_i, 1 \leq i \leq 32$. Each round consists of [13]:

- **addRoundKey.** Given round key $K_i = k^i_{63} \cdots k^i_0$ for $1 \leq i \leq 3$ and current state (STATE) $b_{63} \cdots b_0$, addRoundKey consists of the operation for $0 \leq j \leq 63$:

$$b_j \rightarrow b_j \oplus k^i_j$$

- **sBoxlayer.** PRESENT uses a substitution table (S-box) of 4 bits to 4 bits given in [13]. In this part of the algorithm, the current state $b_{63} \cdots b_0$ is considered as sixteen 4 bit words $w_{15} \cdots w_0$ where $w_i = b_{4*i+3} \| b_{4*i+2} \| b_{4*i+1} \| b_{4*i}, 1 \leq i \leq 15$.
- **pLayer.** In this part, the algorithm perform simple translations of operations by moving the bit number i to the position of $P(i)$ where P is defined by :

$$P(i) = \begin{cases} i * 16 \bmod 63, & i \in \{0, \cdots, 63\} \\ 63, & i = 63 \end{cases} .$$

The Skipjack encryption algorithm is a 64 bits block cipher with 80 bits key length. It performs 32 rounds following two rules named A and B. Each rule represents a series of steps forming a ring according to a linear key permutation G [12]. The algorithm follows the structure of a Feistel network [Fig. 15] and is used in TinySec [14].

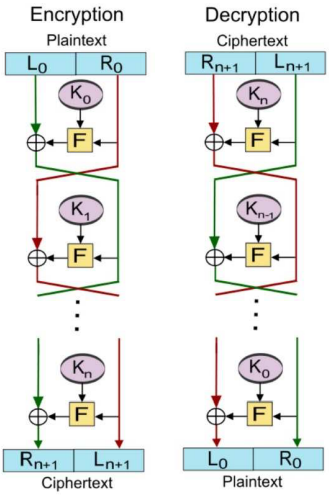


Fig. 15. Feistel Network operations

Rivest Cipher 5 (RC5) is also a block cipher algorithm which works with blocks of 32, 64 and 128 bits. It produces keys length between 40 and 2040 bits and performs in a number of rounds up to 225. Encryption consists of three operations into two registers [12]: an integer addition, a XOR operation and a variable rotation. For encryption, it is suggested to use 64 bits blocks, 20 rounds and 128 bits key

https://sites.google.com/site/ijcsis/
ISSN 1947-5500

length.

There are many other block cipher algorithms such as:

- *Blowfish*: 64 bits blocks, key length from 32 bits up to 448 bits.
- *Twofish*: 128 bits blocks, key length of 128, 192 or 256 bits.
- *Tiny Encryption Algorithm* (TEA): 64 bits blocks, 128 bits key length.
- *Puffin*: 64 bits blocks, 128 bits key length.

As for stream encryption algorithms, there is Rivest Cipher 4 (RC4) which is popular, but it has vulnerabilities and has been the object of successful attacks. Then RC4 should not be used in new applications [15]. For these reasons, the eSTREAM project, part of the ENCRYPT network of excellence, was created to promote new stream cipher systems that can meet both security and constraints related to the software and hardware implementations. The eSTREAM project takes into account two types of profile for different applications:

- *Profile 1* which contains stream ciphers more suitable for software applications with high throughput requirements. This profile portfolio contains the following ciphers: HC-128, Rabbit, Salsa20/12 and SOSEMANUK.
- *Profile 2* stream ciphers which are particularly suitable for hardware applications with restricted resources such as limited storage, gate count, or power consumption [16]. This profile portfolio contains: Grain v1, MICKEY 2.0 and Trivium.

Salsa20/12 stream cipher supports 128 and 256 bits key length. During operation, the algorithm performs 12 rounds and uses bit addition, bit rotation, bitwise XOR and modulo 2^{32} operations. The algorithm is reputed to be simple, scalable and easy to implement on modern microprocessor architectures.

In profile 2, Grain and Trivium are the more accepted ones and have been reported as the most suitable for constrained devices [15]. The initial version of Grain used a key length of 80 bits and an initialization vector of 64 bits. The revised specification, Grain v1, described two stream ciphers: one for 80 bits key length (with an initialization vector of 64 bits) and another for 128 bits key length (with an initialization vector of 80 bits). As for Trivium algorithm, it uses a 80 bits key length and an initialization vector of 80 bits; its secret state has 288 bits, consisting of three interconnected non-linear feedback shift registers of 93, 84 and 111 bits length, respectively [16].

Table II represents a summary of the symmetric encryption algorithms cited in this paper.

Algorithm	Structure	Key size (bits)	round	Cipher type
AES	Substitution and Permutation	128, 192, 256	10, 12, 14	Block
PRESENT	XOR and translation	80, 128	31	Block
Skipjack	Feistel network	80	32	Block
RC5	Rotation	40 to 2040	255	Block
Blowfish	Feistel network	32 to 448	16	Block
Twofish	Feistel network	128, 192, 256	16	Block
TEA	Feistel network	128	64	Block
RC4	-	40 to 2048	256	Stream
Salsa20/12	Rotation and binary operations	128, 256	12	Stream
HC-128	Shift register	128	-	Stream
Rabbit	Shift register	128	-	Stream
SOSEMANUK	Shift register	128 to 256	-	Stream
Grain v1	Shift registers	80 to 128	-	Stream
MICKEY 2.0	Shift registers	80	-	Stream
Trivium	Shift registers	80	-	Stream

TABLE II
SYMMETRIC ALGORITHMS PRIMITIVES

B. Asymmetric Key Encryption Primitives

Asymmetric cryptography, also known as Public Key Cryptography, is deemed as a secure cryptographic system due to the robustness of algorithms. RSA that is used extensively in confidential data exchange is an example of asymmetric cryptography. Due to high demands in terms of calculation capabilities, asymmetric algorithms such as RSA are not applicable in WSN networks which have the constraints of limited processing resources.

Elliptic Curve Cryptography (ECC) is a cryptographic public key system based on mathematics elliptic curves. The main advantage of ECC is the use of small key sizes unlike RSA [17] or ElGamal. It is defined by an equation of the form $y^2 = x^3 + ax^2 + b$ over finite fields F_p (of odd characteristic) or F_{2^m} (of characteristic two) [12]. ECC is based on the discrete logarithm problem which is considered as difficult. Knowing a and c, the problem is to find b such that $a^b = c$ [12]. ECC uses points on an elliptic curve to derive a 160 bits public key which gives the same security level as a 1024 bits RSA key [17].

The Merkle signature scheme (MSS) is a cryptographic system using the hash functions. The system is used to sign a limited number of messages with a single public key *pub*. The number of possible messages must be a power of two, as $2^n, n > 0$. Initially, 2^n public and private keys X_i and Y_i are generated. The algorithm then uses a hash function h to construct a binary tree whose leaves are the values of the function h applied to the 2^n private keys generated. Each node is a concatenation of its two children and the root gives the public key [18]. To sign a message M with MSS,

https://sites.google.com/site/ijcsis/
ISSN 1947-5500

the message M is first signed with a one-time signature scheme and any pair (X_i, Y_i), the result in a signature sig'. Then, for a given leaf i, the algorithm calculates the route $A_0 \dots A_n$ that leads to the root and $auth_j$ is the brother of node A_j in the tree. The signature of M is then given by $sig = (sig' \| auth_2 \| auth_3 \| \cdots \| auth_{n-1})$. The verification of the signature by the recipient will be made following a hash of the public key in sig', followed by a back up to the root via the same path used when signing the message.

The Identity-Based Cryptography (IBC) algorithm introduced by Shamir in 1984 attracted much attention in WSN applications. It eliminates the necessity for checking the validity of certificates in traditional public key infrastructure (PKI). Its main idea is to make an entity's public key directly derivable from its publicly known identity information (e.g. an email address, a telephone number, etc.). Using its master key, a private key generator (PKG) computes then a private key for each user. IBC is particularly suitable for power constrained devices such as sensor nodes. The absence of certificate eliminates the costly certificate verification process [19]. An IBC scheme is composed of four algorithms [20]:

- *Setup* : Given a security parameter $x \in Z_q$ to this algorithm, it outputs system parameters $\mathcal{PP}$ and a master secret M_s. The master secret is only known by the PKG.
- *Key Extraction* : Given a user's identity ID_i and the master secret M_s to this algorithm, it outputs the private key d_{ID_i}. ID_i , d_{ID_i}, and $\mathcal{PP}$ are prestored in a sensor node before deployment.
- *Signature Generation* : Given the message $m \in \mathcal{M}$ and the private key d_{ID_i} , this algorithm outputs the signature σ.
- *Signature Verification* : Given a message $m \in \mathcal{M}$, the signer's identity ID_i, signature σ, and system parameters $\mathcal{PP}$, this algorithm returns valid (1) or invalid (0).

The *Setup* and *Key Extraction* processes are executed before the deployment of the sensor network. The sink node or the base station usually takes the role of a PKG and performs the initialization process through *Setup* and key generation through the *Key Extraction* process.

C. Hash function primitives

A cryptographic hash function H takes an input of variable size and returns a hash value of fixed length while satisfying the properties of preimage resistance, second preimage resistance, and collision resistance [21]. The hash functions such as SHA-1, SHA-2 and MD5 are popular but because of sensors constraints, such solutions are not suitable for WSNs. So, lightweight hash functions based on the PRESENT block construction [13] are proposed for the specific needs of WSN applications:

- The DM-PRESENT hash function takes as input 80 and 128 bit key and returns a 64 bit encryption. The procedure uses a *Devies-Mayer* construction and updates each time a 64 bits chaining variable H_i as $H'_i = E(H_i M) \oplus H_i$ [21], when E is a PRESENT encryption.
- The H-PRESENT hash function takes as input two 64 bit variables H_1, H_2 and a 64 bit block message M; noted by the triplet (H_1, H_2, M). The function returns a pair (H'_1, H'_2) when $H'_1 = E(H_1, H_2 \| M) \oplus H_1$ and $H'_2 = E(H_1 \oplus c, H_2 \| M) \oplus H_1$ [21]. E is a 128 bit encryption with PRESENT and c a nonzero constant.

The SQUASH algorithm is another hash function primitive which is proposed by Shamir in [22] for RFID tags by adopting a challenge-response authentication. In this protocol, the tag and the reader share a secret key S. The reader issues a random challenge R, and the tag responds with the value $H(S, R)$ where H is some publicly known hash function. This value is independently computed by the reader, which accepts the authentication if and only if the computed and received values are the same [22].

V. Authentication schemes in WSNs

WSNs collect environmental data, process and store, and finally send the data to users on demand or following detection of an event [23]. This underlies that the WSN offer services to the users, hence the need to define user authentication methods. In the other hand, hackers can easily inject their own data in a sensor network. To ensure data security, the sensors must authenticate data they received from a source before any operation. Furthermore, authentication schemes enable a recipient to verify that data was sent from a trusted source [24]. In addition to user authentication protocols, node authentication protocols must also be integrated into WSN, in order to ensure the authenticity of new nodes which need to access to the network. Sensor networks are comprised of hundreds of nodes that are dispersed in the nature to collect information. Because of their limited resources or attacks, some sensors may be lost during the evolution of the network. To extend the life of the network, the deployment of new authenticated nodes becomes necessary [25].

A. Dynamic user authentication based on hash functions

In [26], Wong et *al.* propose a dynamic user authentication scheme based on hash functions. The protocol consists of three phases: Registration, Login and Authentication. The protocol allows any legitimate user to query data from all sensors. Furthermore, only simple operations are used during the three phases. This helps in optimizing sensor resources.

The Tseng et *al.* [27] scheme is an improvement of Wong et *al.* scheme and takes place over four phases : Registration, Login, Authentication and Password change. The authors demonstrate that the Wong et *al.* scheme has some vulnerability such as replay attacks, forgery attacks, too much trust granted to the gateway node and no password change procedure. This scheme does not only fix the weaknesses but also enhances the security of Wong et *al.* scheme by allowing a legitimate user to change their password.

https://sites.google.com/site/ijcsis/
ISSN 1947-5500

As for Vaidya et *al.*, they propose a user authentication scheme [28] that takes place on four phases: Registration, Login, Authentication and Password changing. They show some vulnerabilities of Tseng et *al.* scheme related to the replay and man-in-the-middle attacks. Their scheme adds protections against the replay, forgery and man-in-the-middle attacks but also provide mutual authentication between regular nodes and gateway nodes [28].

In [29], Faye et *al.* propose another authentication scheme based on a probabilistic analysis of the DoS attack risk. They show some vulnerability of Vaidya et *al.* scheme related to the DoS and timestamp forgery attacks and then propose a solution for solving these weaknesses. Such as the Vaidya et *al.* scheme, this solution takes place on the same phases.

B. $\mu TESLA$: an authentication scheme based on message broadcasting

Perrig et *al.* [24] propose a broadcast message authentication scheme named $\mu TESLA$. The difficulty of applying a digital signature in WSNs led the authors to adopt the symmetrical cryptography combined to an asymmetric approach. $\mu TESLA$ requires that the base station and the regular nodes are time synchronized. To send an authenticated packet, the base station simply computes a message authentication code (MAC) on the packet with a key which is secret at this moment. When a node receives a packet, it can verify that the corresponding MAC key was not yet disclosed by the base station (based on clock synchronization, its maximum synchronization error, and the time schedule at which keys are disclosed) and stores the packet in a buffer. When a node receives the disclosed key, it can easily verify the correctness of the key. If the key is correct, the node can now use it to authenticate the previous packet stored in its buffer.

In [30], Liu et *al.* propose a practical broadcast authentication scheme based on Merkle tree to address weaknesses of $\mu TESLA$ by solving the problems of scalability and DoS attacks. In $\mu TESLA$, during the key disclosure period, a node can be flooded with fake packets that it store in its buffer. In this algorithm [30], any delayed packet in the authentication process is deleted in order to prevent from DoS attacks. This solution also allows multiple simultaneous transmissions, unlike $\mu TESLA$ in which only the base station is supposed to broadcast the messages.

C. Mutual authentication and key agreement schemes

A user authentication and key agreement scheme is described in [31] where Turkanovìc et *al.* propose a solution for heterogeneous networks composed of a gateway node and sensors nodes. The scheme ensures significant features like mutual authentication, password security, single registration, key agreement, high security and low computational costs. They use symmetric cryptography combined with hash functions to optimize sensor resources. It takes place in five phases: Pre-deployment, Registration, Login, Authentication and Dynamic node addition.

Amin and Biswas [32] pointed out some vulnerabilities of Turkanovìc et *al.* scheme such as off-line identity guessing attack, off-line password guessing attack, node impersonation attacks and inefficiency in the authentication phase. They present a novel architecture based on remote user authentication and key agreement protocol for nodes which want to access the network. The protocol takes place in several phases such as System setup, User registration, Login, Authentication, Password update and Dynamic node addition. It claims to resolve Turkanovìc et *al.* security weaknesses.

In [33], Lu et *al.* show the vulnerability of the Amin and Biswas scheme related to the eavesdropping attack and then propose a mutual authentication and key agreement that prevents an adversary to get sensitive data. Their protocol provides mutual authentication and is claimed to be resistant to insider attacks, DoS attacks and sensor node impersonation attacks. It consists of a five phases: User registration, Sensor node registration, Login, Authentication and key agreement and Password change.

D. Access control in wireless sensor networks

Zhou et *al.* propose an access control scheme based on ECC [25]. The protocol restricts access to the network for malicious nodes and runs on two main phases: a pre-deployment phase and an authentication phase. The scheme uses a certificate authority (CA) that attribute certificates to the new nodes which need to access to the network. In the certificate, the node ID information and its bootstrapping time are signed by CA's private key to add integrity protection, so that adversaries cannot falsify the ID and the bootstrapping time. When the new node is deployed into the sensor network, it can show its certificate to its neighbors. The neighbors can verify the ID and the bootstrapping time with the CA's public key. A new node can be accepted into the sensor network if and only if it has a correct identity and its bootstrapping time is within a tolerance period of current time.

In [34], Huang propose an access control protocol using the concept of time bound in which, once time period has elapsed, the sensor node cannot access any data for future time period. Unlike the Zhou et *al.* scheme in which the authors assume that each sensor node can sustain a tolerance time interval before it is compromised, this scheme delete this assumption by dividing the time duration of the network into t periods, numbered $1, 2, \cdots, t$. The scheme uses ECC and is resistant to the Sybil and Wormhole attacks.

Table III shows for each authentication scheme its vulnerabilities in relation to some of the attacks presented in the section III. In this table, the X indicates that the scheme does not withstand the attack in question.

https://sites.google.com/site/ijcsis/
ISSN 1947-5500

Scheme	Sybil	Wormhole	Sinkhole	Hello Flood	Select. Forw	Jamm.	Replay	Node Capt.
Perrig et *al.*[24]				X		X		
Zhou et *al.*[25]	X	X		X		X		
Wong et *al.*[26]				X		X	X	X
Tseng et *al.*[27]				X		X	X	
Vaidya et *al.*[28]				X		X		
Faye et *al.*[29]				X		X		
Liu et *al.*[30]				X		X		X
Turkanovic̀ et *al.*[31]	X			X		X	X	X
Amin-Biswas[32]				X		X	X	X
Lu et *al.*[33]				X		X		X
Huang et *al.*[34]	X			X		X		X

TABLE III
AUTHENTICATION SCHEME VULNERABILITIES RELATED TO SOME OF THE ATTACKS.

VI. SECURING NETWORK OPERATIONS

In this section we discuss about some of the many principles of securing network operations such as the neighborhood discovery process, routing protocols, clustering schemes, redundancy and load balancing techniques, data partitioning.

A. Security of the neighborhood discovery process

One of the main advantages of the WSNs is the ability of the nodes to self-organize in a hostile environment to collect the data. The discovery of neighbors is often a preliminary step for most network operations. The neighborhood of a node X is considered as being the set of the nodes which are located in the transmission range of the node X. The security of the neighborhood discovery process is very important because an attacker can disrupt this operation by causing two nodes, which are not in the same neighborhood, to believe that they are neighbors or by preventing two real neighbor nodes from becoming neighboring. Many neighborhood discovery security protocols are proposed. The basic approach is the use of distance in which the distance between two nodes is calculated by measuring the crossing time of the signal between the two nodes. These two nodes are considered to be neighbors if the computed distance is less than the radio signal of the nodes.

Wormhole attacks are the main threats of neighbor discovery protocols. An attacker can easily implement a wormhole attack without the need of breaking cryptographic keys. So, in [35], the authors propose a neighbor discovery scheme based on localization which resists wormhole attack.

SEDINE [36] is another protocol for securing the neighborhood discovery operations. Based on the listening of communications between nodes, SEDINE relies on two phases: neighbor discovery and neighbor verification. The SEDINE is described as a good scheme which can be applied in routing operations. An analysis done in [37] shows that this protocol prevents two non-legitimate neighbors from being considered as neighbor nodes. Nevertheless its main drawbacks are the fact that it is vulnerable to Sybil attack, and it does not allow to protect itself when a malicious node prevents two legitimate neighbor nodes from becoming neighbors [37].

B. Securing routing protocols

Most of the proposed routing protocols aim to optimize the resources of the sensors and are not security-oriented. With the many attacks that can arise during the routing process, it is essential to set up reliable means of securing routing protocols. In [38], Deng and $al.$ propose a secure routing protocol called INSENS. The main idea is to allow the base station to construct for each node a routing table which will then be transmitted to the node in a secure manner. This protocol is based on three principles [38]:

- using of multiple paths in order to be resistant to attacks,
- optimizing the resources of the nodes by computing all the heavy operations at the level of the base station which holds greater capacities than the other sensors,
- limiting the attacks by adopting authentication mechanisms with symmetric keys between the base station and the other nodes.

To prevent the WSN from DoS attacks, no node is allowed to transmit messages on the entire network; only the base station is authorized to do so. The base station uses a hash function to generate a key sequence to avoid incorrect routing paths that may be caused by a malicious node. The protocol takes place in two phases: the paths discovery and the data transmission.

INSENS has many advantages for secure routing in WSNs. However, its main disadvantage is the fact that the base station fulfills many responsibilities. Then compromising of such a node will allow an attacker to gain a total control of the network. Furthermore, it can happen that only one route exists between two nodes and in this case compromising this route could disrupt communications.

As for the ARIADNE protocol [39], Hu and $al.$ propose on demand routing protocol in which a node proceeds to the route discovery if and only if it has packets to transmit [39]. ARIADNE relies on the $TESLA$ authentication protocol to guarantee the authenticity and the freshness of the data. Moreover, this scheme prevents from DoS attacks and nodes compromising, and then guarantees the efficiency of paths discovery. The ARIADNE protocol is based on symmetric cryptography to ensure security of communications. It takes place on two phases: the paths discovery phase and the routing maintenance phase. During the discovery process, each node

https://sites.google.com/site/ijcsis/
ISSN 1947-5500

adds its own address and a calculated message authentication code (MAC) with its key. This ensures the integrity by preventing an attacker from modifying the routing data.

C. Clustering formation process

Clustering in wireless sensor networks is a means of identifying a subset of nodes in the network and assigning it a cluster head that will be responsible for specific tasks such as routing or data aggregation process. Thus it makes it possible to reduce the complexity of the routing algorithms, to optimize the medium resource, to facilitate data aggregation, to simplify network management (in particular address assignment and cryptographic keys management), to optimize energy expenditure, and finally to make the network more scalable.

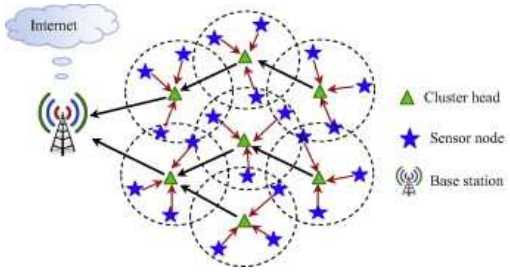


Fig. 16. WSN Clusters

As clustering is an effective means of reducing the energy consumption of the sensors, many cluster based routing protocols are proposed. In addition to attacks against routing protocols, clustering protocols may also be vulnerable to other types of attacks. An attacker may attempt to disrupt the clusters formation process in such a way that some nodes integrate inappropriate clusters. Moreover, a compromised cluster head could disrupt network operations: it can choose to relay data to a bad recipient; it can also choose not to relay the packets; it can also choose to send falsified data toward the base station, and so on. Therefore it is essential to take into account security to form appropriate and energy efficient clusters. In [40], Zhang and al. propose a key management scheme designed for dynamic cluster formation. Their algorithm uses a hash function and is resistant against attacks on cluster heads. This mechanism also includes, a phase of updating the keys after a certain time. Another WSN key management protocol is also described in [41]. For this last one, three types of keys are used: a secret shared key between a pair of regular nodes, a secret shared key between a cluster head and each node within its cluster, and a third secret shared key used for communications between a cluster head and any other given node.

D. Redundancy and load balancing techniques

Redundancy and load balancing mechanisms are primary solutions that respond to the availability security requirement. Often used in routing and clustering algorithms, these techniques significantly improve the network efficiency by extending its lifetime and decreasing the overall packet loss rate. The load balancing also helps evenly splitting the load on all nodes in the WSN. So there are major efficiency benefits that stem from load balancing which helps in guaranteeing a better quality of service by increasing the availability and the capacity of the network [42]. For example, using multiple redundant routes between a source and a destination will require the attacker to explore all possible routes to be able to discover the complete information.

E. Data partitioning

Data partitioning is a solution to prevent the retrieval of information in the WSN by partitioning data. If a sensor has to send information, it will cut it into several fixed small packet sizes. These packets will be sent on different paths toward the base station that collects them in order to reconstruct the original information. This mechanism obliges an attacker to retrieve all the packets if he wants to retrieve the information. He must also be able to listen to the whole network, to recover the different packets which pass through different paths. However, this solution has the drawback to considerably increase energy consumption because it involves more transmissions and a large number of nodes for routing information (split into small packets) to the base station.

VII. Conclusion

Constraints on the sensors naturally lead to specific attacks aimed at exploiting the vulnerabilities of the nodes and the network. Moreover, sensor networks are also subject to most of the attacks carried out in other conventional networks. In this paper, we have reviewed attacks on WSNs, before presenting few security solutions. Authentication is an important security requirement in most of the applications and especially in wireless sensor networks. In WSN, one of the main goals is to guarantee access to the network only for legitimate nodes while prohibiting malicious nodes from entering it. Furthermore, mechanisms that ensure data integrity in the network help to have a greater accuracy in order to taking good decisions.

In a more general way, related security issues in WSN are numerous and complex. This complexity is most often linked to the constraints on the sensors. Indeed, the sensors are characterized as entities that have low memory storage, low computing power and low CPU performance. Because of these limitations, classical cryptographic solutions which require many calculations in the encryption and decryption operations become unsuitable for this type of network. That is why there are works on lightweight cryptography, which concerns the science of small cryptography keys. Nevertheless, as we can see in this report, it is still found that many works continue offering relatively long cryptography keys with respect to the size of the protected data without caring about energy over-consumption [43].

Finally, it is also noted that most of the existing solutions remain vulnerable to certain attacks.

https://sites.google.com/site/ijcsis/
ISSN 1947-5500

ACKNOWLEDGMENT

The authors would like to thank "Centre d'Excellence en Mathématiques, Informatique et TIC (CEA-MITIC)" ; UFR Sciences appliquées et de Technologies (UFR SAT), Université Gaston Berger, Saint-Louis, Sénégal. (http://www.ceamitic.sn/).

REFERENCES

[1] "Wsn architecture illustration: http://vlssit.iitkgp.ernet.in/ant/ant/8/theory/."
[2] I. Kamel and H. Juma, "A lightweight data integrity scheme for sensor networks," *MDPI*, 2011.
[3] M. H. Eldefrawy, M. K. Khan, and K. Alghathbar, "Broadcast authentication for wireless sensor networks using nested hashing and the chinese remainder theorem," *MDPI*, 2010.
[4] H. M. Alzaid, "Secure data aggregation in wireless sensor networks," Ph.D. dissertation, Queensland University of Technology, 2011. [Online]. Available: http://eprints.qut.edu.au/41843/
[5] C. Kaufman, R. Perlman, and M. Speciner, *Network Security: Private Communication in a Public World, 2nd Edition*. Prentice Hall, 2002.
[6] M. A. Salam, "Detectability of man-in-the-middle attacker in mobile sensor networks," *International Journal of Advanced Computer Science, Vol.3, No.2, Pp.99-103, Feb., 2013*, 2013.
[7] S. Glass, V. Muthukkumurasamy, and M. Portmann, "Detecting man-in-the-middle and wormhole attacks in wireless mesh networks," *International Conference on Advanced Information Networking and Applications*, 2009.
[8] K.Venkatraman, J. Daniel, and G.Murugaboopathi, "Various attacks in wireless sensor network : Survey," *International Journal of Soft Computing and Engineering (IJSCE)*, vol. 3, 2013.
[9] S. Marano, V. Matta, and L. Tong, "Distributed inference in the presence of byzantine sensors."
[10] L. Lamport, R. Shostak, and M. Pease, "The byzantine generals problem, acm transactions on programming languages and systems, vol. 4, no. 3, pp. 382–401, july 1982."
[11] N. Alajmi, "Wireless sensor networks attacks and solutions," *(IJCSIS) International Journal of Computer Science and Information Security, Vol. 12, No. 7, July*, 2014.
[12] R. Roman, C. Alcaraz, and J. Lopez, "A survey of cryptographic primitives and implementations for hardware-constrained sensor network nodes," October 2007.
[13] A. Bogdanov, L. R. Knudsen, G. Leander, C. Paar, A. Poschmann, M. J. B. Robshaw, Y. Seurin, and C. Vikkelsoe, "Present: An ultra-lightweight block cipher," in *THE PROCEEDINGS OF CHES 2007*. Springer, 2007.
[14] X. Zhang, H. M. Heys, and C. Li, "Energy efficiency of symmetric key cryptographic algorithms in wireless sensor networks."
[15] C. Manifavas, G. Hatzivasilis, K. Fysarakis, and K. Rantos, "Lightweight cryptography for embedded systems - a comparative analys," *Springer*.
[16] "eSTREAM web site : http://www.ecrypt.eu.org/stream/index.html."
[17] Arpit and A. Kumar, "Optimized elliptic curve cryptography as fine balance for wireless sensor network," *International Journal of Modeling and Optimization*, 2011.
[18] G. Becker, "Merkle signature schemes, merkle trees and their cryptanalysis," 2008.
[19] C. Cheng-Kang, L. J. K., Z. Jianying, B. Feng, and D. R. H., "Practical id-based encryption for wireless sensor network," in *Proceedings of the 5th ACM Symposium on Information, Computer and Communications Security*, ser. ASIACCS '10. New York, NY, USA: ACM, 2010, pp. 337–340. [Online]. Available: http://doi.acm.org/10.1145/1755688.1755734
[20] H. K. Patil and S. A. Szygenda, *Security for Wireless Sensor Networks using Identity-Based Cryptography*. CRC Press, 2012.
[21] B. Andrey, L. Gregor, P. Christof, P. Axel, R. M. J. B., and S. Yannick, "Hash functions and rfid tags: Mind the gap." in *CHES*, ser. Lecture Notes in Computer Science, E. Oswald and P. Rohatgi, Eds., vol. 5154. Springer, 2008, pp. 283–299. [Online]. Available: http://dblp.uni-trier.de/db/conf/ches/ches2008.htmlBogdanovLPPRS08
[22] S. Adi, "Squash - a new mac with provable security properties for highly constrained devices such as rfid tags." in *FSE*, ser. Lecture Notes in Computer Science, K. Nyberg, Ed., vol. 5086. Springer, 2008, pp. 144–157. [Online]. Available: http://dblp.uni-trier.de/db/conf/fse/fse2008.htmlShamir08
[23] Z. Benenson, N. Gedicke, and O. Raivio, "Realizing robust user authentication in sensor networks," in *Real-World Wireless Sensor Networks (REALWSN)*, 2005.
[24] A. Perrig, R. Szewczyk, V. Wen, D. Culler, and J. D. Tygar, "Spins: Security protocols for sensor networks," in *Wireless Networks*, 2001, pp. 189–199.
[25] Y. Zhou, Y. Zhang, and Y. Fang, "Access control in wireless sensor networks," *Ad Hoc Networks*, pp. 3–13, 2007.
[26] W. K. H. M., Z. Yuan, C. Jiannong, and W. Shengwei, "A dynamic user authentication scheme for wireless sensor networks." in *SUTC (1)*. IEEE Computer Society, 2006, pp. 244–251. [Online]. Available: http://dblp.uni-trier.de/db/conf/sutc/sutc2006-1.htmlWongZCW06
[27] T. Huei-Ru, J. Rong-Hong, and Y. Wuu, "An improved dynamic user authentication scheme for wireless sensor networks." in *GLOBECOM*. IEEE, 2007, pp. 986–990. [Online]. Available: http://dblp.uni-trier.de/db/conf/globecom/globecom2007.htmlTsengJY07
[28] B. Vaidya, M. Chen, and J. J. P. C. Rodrigues, "Improved robust user authentication scheme for wireless sensor networks," in *Wireless Communication and Sensor Networks (WCSN), 2009 Fifth IEEE Conference on*, Dec 2009, pp. 1–6.
[29] F. Youssou, Guyennet, Hervé, and N. Ibrahima, "Authentification robuste basée sur une analyse probabiliste de risque d'attaque par déni de service dans les réseaux de capteurs sans fil," in *CARI 2012, 11ème Colloque Africain sur la Recherche en Informatique et en Mathématiques Appliquées*, Alger, Algeria, Oct. 2012.
[30] D. Liu, P. Ning, S. Zhu, and S. Jajodia, "Practical broadcast authentication in sensor networks," in *2nd Annual International Conference on Mobile and Ubiquitous Systems (MobiQuitous 2005), 17-21 July 2005, San Diego, CA, USA*, 2005, pp. 118–132. [Online]. Available: http://dx.doi.org/10.1109/MOBIQUITOUS.2005.49
[31] M. Turkanović, B. Brumen, and M. Hölbl, "A novel user authentication and key agreement scheme for heterogeneous ad hoc wireless sensor networks, based on the Internet of Things notion," *Ad Hoc Networks*, vol. 20, pp. 96 – 112, 2014. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S157087051400064X
[32] A. Ruhul and B. G.P., "A secure light weight scheme for user authentication and key agreement in multi-gateway based wireless sensor networks," *Ad Hoc Netw.*, vol. 36, no. P1, pp. 58–80, Jan. 2016. [Online]. Available: http://dx.doi.org/10.1016/j.adhoc.2015.05.020
[33] Y. Lu, Li, Lixiang, Peng, Haipeng, and Y. Yang, "An energy efficient mutual authentication and key agreement scheme preserving anonymity for wireless sensor networks," *Sensors*, vol. 16, no. 6, p. 837, 2016. [Online]. Available: http://www.mdpi.com/1424-8220/16/6/837
[34] H. Hui-Feng, "A new design of access control in wireless sensor networks." *IJDSN*, vol. 2011, 2011. [Online]. Available: http://dblp.uni-trier.de/db/journals/ijdsn/ijdsn2011.htmlHuang11
[35] M. G. Otero and A. Poblacion-Hernandez, "Secure neighbor discovery in wireless sensor networks using range-free localization techniques," *International Journal of Distributed Sensor Networks (IJDSN)*, 2012.
[36] S. Hariharan, N. B. Shroff, and S. Bagchi, "Secure neighbor discovery through overhearing in static multihop wireless networks," *Computer Networks, 55(6):1229-1241*, 2011.
[37] S. Bagchi, S. Hariharan, and N. B. Shroff, "Secure neighbor discovery in wireless networks," *Electrical and Computer Ingineering*, 2007.
[38] J. Deng, R. Han, and S. Mishra, "Insens: Intrusion-tolerant routing in wireless sensor networks," *Published by Elsevier B.V*, 2005.
[39] Y.-C. Hu, A. Perrig, , and D. B. Johnson, "Ariadne: A secure on-demand routing protocol for ad hoc networks," *Wirel. Netw., 11(1-2) :21-38, January*, 2005.
[40] P. J. Y. Zhang, B. Zheng, and J. Co, "A key management method based on dynamic clustering for sensor networks," *International Journal of Distributed Sensor networks*, 2015.
[41] X. Yu, "Security key management scheme in heterogeneous wireless sensor networks based on clustering strategy," *JNW, 8(9) :2114-2120*, 2013.
[42] C. Diallo, M. Marot, and M. Becker, "Efficiency benefits through load-balancing with link reliability based routing in wsns," *In International Journal on Advances in Networks and Services, Vol 3 n34, Feb. 2011*, 2011.
[43] M. T. Sow and C. Diallo, "Energy over-consumption induced by securing network operations," *2nd IEEE International Conference on Frontiers of Sensors Technologies, Shenzhen, China, (IEEE ICFST 2017).*, April, 2017.

https://sites.google.com/site/ijcsis/
ISSN 1947-5500