Disponible en www.sciencedirect.com

Journal of Applied Research and Technology

Journal of Applied Research and Technology 13 (2015) 342-350

CCADET CENTRO DE CIENCIAS APLICADAS Y DESARROLLO TECNOLÓGICO

www.jart.ccadet.unam.mx

Original

Secure collaborative system in heterogenous wireless sensor networks

M. Kasraoui*, A. Cabani, H. Chafouk

Research Institute on Embedded Electronic Systems (IRSEEM), IIS Group, St-Etienne-du-Rouvray, France

Received 19 April 2014; accepted 18 August 2014

Abstract

The IPv6 over Low power Wireless Personal Area Networks (6LoWPANs) have turned out to be one of the most emerging field in Wireless Sensor Networks (WSNs) which can be integrated with Internet technology. 6LoWPAN network consists of heterogeneous wireless sensors which have high resource-constraints such as bandwidth, processing power, memory, energy, etc. The resource-constraints put forth many challenges to apply the available standard security protocols such as Transport Layer Security (TLS), Internet Protocol Security (IPSec), Internet Key Exchange version 2 (IKEv2), etc., for the interconnection of Heterogeneous Wireless Sensor Networks (HWSNs) with Internet. To overcome these situations, the researchers aimed to reinforce and adapt the end-to-end security between Internet and the IP enabled sensor networks. The above mentioned security protocols are not modified at the Internet end point in HWSNs. Hence we are proposing a novel Cooperative Key Exchange System (CKES) by using the concept of Chinese Remainder Theorem (CRT). We have used NS2 simulator to implement the proposed concept and also compared with IKEv2.

Keywords: IPsec; 6LoWPAN; IKEv2; Security; HWSNs

1. Introduction

Over the last decade, HWSNs are used as a suitable solution for a large number of application scenarios like logistic, military, health, etc. The heterogeneity considered characterizes sensor nodes (SNs) by their diversity in terms of calculation, link and energy. With the development of technology, many standardization protocols such as WirelessHART, ISA 1000.11.a, ZigBee, etc., are focusing their efforts to implement a global network infrastructure (Granjal et al., 2008; Granjal et al., 2010a). This allows the SNs to communicate directly with the compatible IPv6 hosts.

At the beginning, IETF (Internet Engineering Task Force) group proposed an adaptation layer called 6LoWPAN as mentioned in Figure 1. This involves the transmission of IPv6 packets over IEEE 802.15.4 network (IEEE std. 802.15.4, 2003) in an appropriate way in terms of power consumption, memory usage and packet size.

The 6LoWPAN (Montenegro et al., 2007) adaptation layer is located between the network and the link layers. It provides header compression and packet fragmentation functionality for IPv6 packets (Yu et al., 2013).

In this paper, we have proposed a new architecture to develop and refine an IP security solution for HWSNs in the context of "Port Transit of Containers" project. This project aims to develop a system to trace the mobility of containers, material types inside the containers, delivering address, etc., by using the HWSNs technologies with utmost security. Nevertheless, the interconnection between HWSNs and the Internet requires secure and safe communication. From the IPv6 hosts side, the IPSec protocol is supported to secure the end-to-end communication. From the 6LoWPAN mote side, there is no standardized solution offering the end-to-end security unlike secure routing standards and protocols TinySec (Karlof et al., 2004) and MS-SPIN (Dhurandher et al., 2010). Hence, it would be interesting to adapt the IPSec protocol in order to ensure an optimized security in constrained environments. The most important IPsec feature is the encryption and the authentication of the end-to-end traffic at the IP level. It should be possible to secure all applications by just turning on the IPsec.

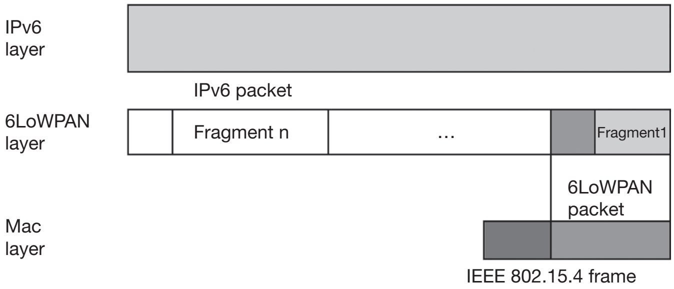


Fig. 1. 6LoWPAN adaptation layer.

*Corresponding author.
E-mail address: mo.kasraoui@esigelec.fr (M. Kasraoui).

IPsec defines an Authentication Header (AH) and an Encapsulating Security Payload (ESP) (Manral, 2007). The AH can be used to provide data integrity and authentication while ESP provides data confidentiality. IPSec secured links are defined in terms of Security Associations (SAs). Each SA is maintained between two or more entities which describe the algorithms, keys and other security parameters to be used. To ensure a dynamic management of security associations, an IKEv2 protocol was defined in RFC 5996 (Kaufman et al., 2010) which uses two databases Security Association Database (SAD) and Security Policy Database (SPD). These databases are used to store all security associations and policies for each device. Four pairs of messages are needed to negotiate one security association. These requirements present challenges for the implementation of IKEv2 on wireless environments by considering the processor cost and bandwidth limitation. So, there is need to develop a lightweight IKE which can be easily deployed in the target network.

This paper is organized as follows: section 2 gives a summary of related works for the proposed approach. Section 3 describes the IKEv2 protocol, AH and ESP. Section 4 describes our proposed approach. Section 5 gives the simulation results. The conclusions and future works are discussed in section 6.

2. Related work

This section presents an overview on several works related to the security problems in IP-enabled WSNs. The related works which we have carried out mainly focuses on how to ensure the IP communication end to end security.

Granjal et al. (2010b) proposed a Secure Interconnection Model for WSN (SIMWSN) that provides confidentiality, authentication and integrity. SIMWSN is based on a security gateway to filter and secure IP communications between the Internet and the sensor nodes. It also establishes an indirect connection between them to protect WSN against Internet attacks. In SIMWSN, each Internet host should use IPSec in tunnel mode to connect the security gateway. However, on WSN side, the basic IEEE 802.15.4 security mechanisms are used. Figure 2 shows how to employ secure gateways in order to associate all WSNs. In WSN, SIMWSN applies the rules defined in 6LoWPAN (Montenegro et al., 2007) to manage the IPv6 addresses, and support both IPv6 and 6to4 tunneling on the internet interface.

The concept of SIMWSN has not been implemented in the above mentioned works. Raza et al. (2011) have followed the idea of using network-layer security in IP-enabled WSNs. The authors have implemented a compressed IPsec for 6LoWPAN

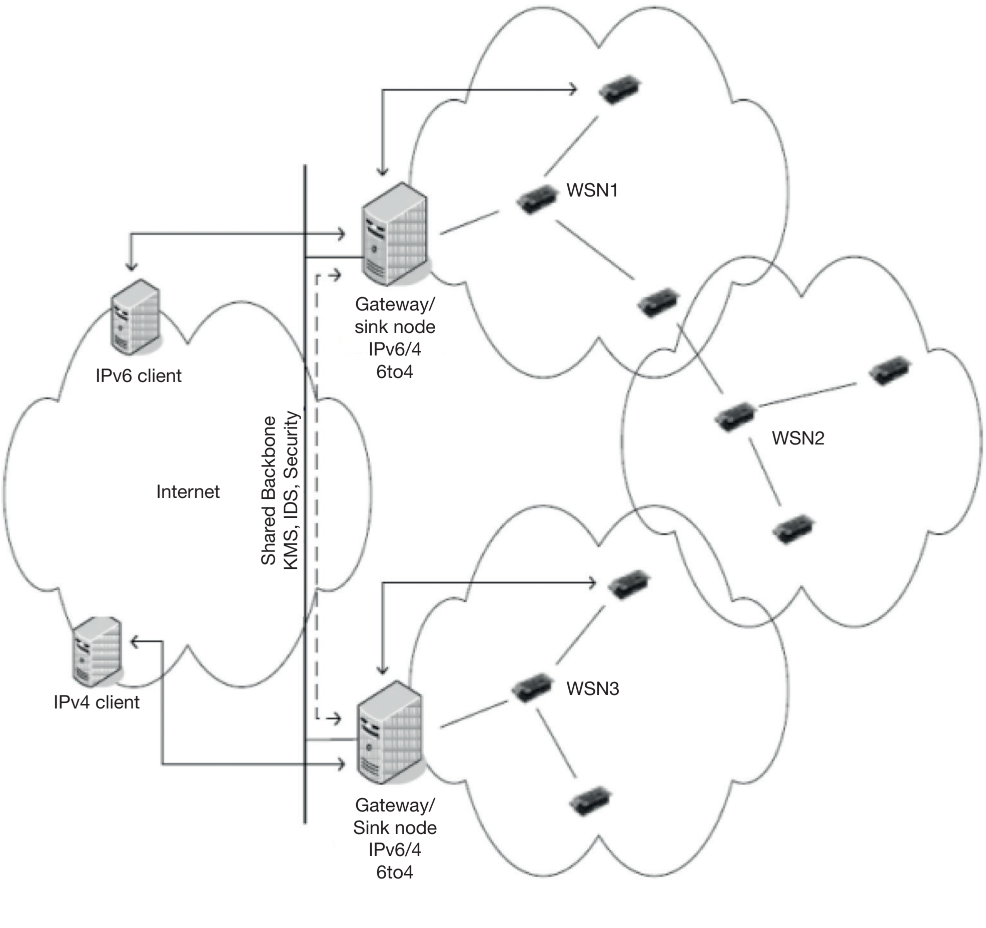


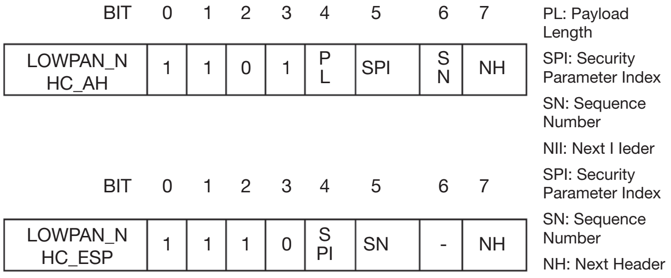
Fig. 2. Illustrates the operational scenario of SIMWSN [1].

networks. They have also developed an encoding method for the AH and the ESP extension headers using the LOWPAN Next Header Compression (NHC) format introduced in Hui and Thubert (2011). Figure 3 shows how encoding has been done in AH and ESP headers.

Raza et al. (2011) have implemented a compressed version of IPSec in the Contiki OS using pre-shared key concept to establish Security Association (SA). According to the obtained results (Raza et al., 2011), the overall memory footprint of the IPsec implementation ranged from 3.9 kB to 9 kB ROM and 0.3 kB to 1.1 kB RAM depending on the protocol used and the mode of operation (HMAC-SHA1-96 for AH and AES-CBC for ESP).

Gupta et al. (2005) have proposed Sizzle which provides gateways with the use of SSL protocol and a proprietary communication on the WSN. Woo Young Jung et al. (2009) have proposed a SN for an All-IP World (SNAIL) which employs the same cryptographic concept as in Gupta et al. (2005), but without using security gateway. Casado and Tsigas (2009) have proposed ContkiSec, which adds the usage of security profiles for WSN. Table 1 shows a comparison study of the about discussed solutions in terms of security characteristics.

Most researchers have focused on the use of IPSEC. The proposed idea of Raza et al. (2012) brings our attention to adapt IPSec in 6LoWPAN. This solution has some disadvantages such as the use of conventional IKE without adapting it in the WSN constrained environments. In order to improve secure and efficient key management in large-scale HWSN, we have proposed a novel Cooperative Key Exchange System (CKES). This incorporates the best features of IKEv2 taking in consideration of constraint SNs. We previously presented a formal analysis in Kasraoui et al. (2014a) of CKES security. Hence, in this paper, our work is dealing with the energy consumption of CKES.



BIT	0	1	2	3	4	5	6	7
LOWPAN_NHC_AH	1	1	0	1	PL	SPI	SN	NH

BIT	0	1	2	3	4	5	6	7
LOWPAN_NHC_ESP	1	1	1	0	SPI	SN	-	NH

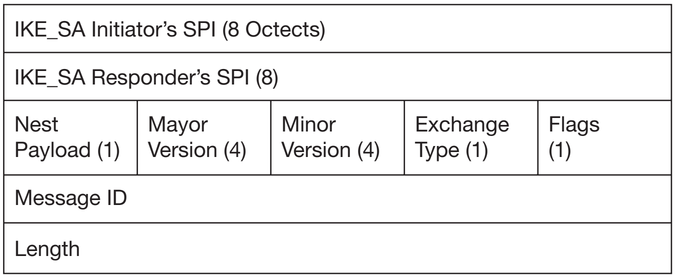
PL: Payload Length
SPI: Security Parameter Index
SN: Sequence Number
NII: Next I leder
SPI: Security Parameter Index
SN: Sequence Number
NH: Next Header

Fig. 3. NHC encoding for IPv6 AH and ESP Extension Headers. NH: next header; PL: payload length; SN: sequence number; SPI: security parameter index.

3. Internet key exchange IKEv2

The IPSEC (Kent & Seo, 2005) secured links are defined in terms of SAs. Each SA is maintained between two or more entities that describe the cryptography algorithms, keys and other security parameters. To ensure a dynamic management of security associations an IKEv2 protocol was defined in RFC 5996 which uses two databases SAD and SPD. These databases are used to store all security associations and the security policies for each device. Figure 4 shows the IKEv2 header information.

In IKEv2, all communications consist of pairs of messages which are called "request/response pairs". To maintain a security association, two phases are required by IKEv2. Phase 1 in Figure 5 performs mutual authentication between two parts and establish an IKE_SA.



IKE_SA Initiator's SPI (8 Octects)				
IKE_SA Responder's SPI (8)				
Nest Payload (1)	Mayor Version (4)	Minor Version (4)	Exchange Type (1)	Flags (1)
Message ID				
Length				

Fig. 4. IKE header format.

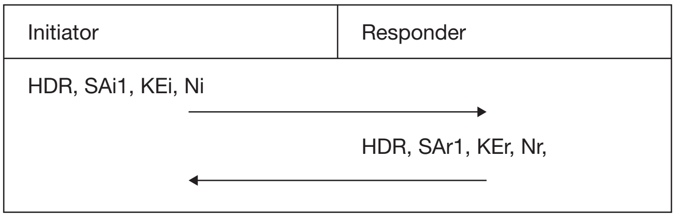


Fig. 5. IKEv2 Pahse 1 exchange.

Table 1
Proposals for IP communication end-to-end security on WSNs.

	SSNAIL	Sizzle	ContikiSec	SIMWSN	6LoWPAN/IPSec
Authentication	ECDSA	ECDSA	CMAC	ECDSA	AH-HMAC-SHA1-96
Key exchange	ECDH	ECDH	—	ECDH	ISAKMP/ECDH
Confidentiality	RC4	RC4	AES-CBC	AES/CCM or 3DES	AES-CBC, AES_CTR, 3DES
Key size	160	160	128	128->256	128 -> 196
Hashing	MD5,SHA1	MD5,SHA1	—	SHA1,SHA2	SHA1, Tigger (x3SHA1)
Access control	—	Gateway	—	Gateway	—
Layer	Transport	Transport	MAC/ Network	Network	Network
Gateway	—	Yes	—	Yes	—
End-to-end security	Yes (transport)	Yes (transport)	No	Yes (Network)	Yes (Network)
Attacks	MIM	MIM	Eavesdropping, Replay, DoS,	Replay,	MIM, Spoofing(UI), DoS, Replay, Black hole,
Network layer	—	—	—	IPSec_TM/ WSN_SM	IPSec_PAN

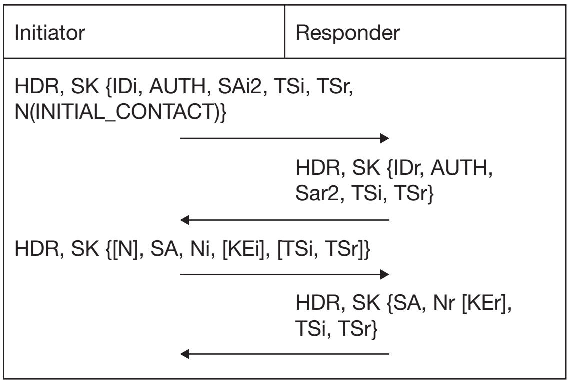
At this point, both nodes have a shared secret to perform encryption and integrity protection for further IKEv2 exchanges. They will be agreed each other on the following parameters of their IKE_SA:

- Cryptographic algorithms: algorithms to protect IKE exchanges, Diffie-Hellman Groups (Group 1: 768-bit MODP, and Group 2: 1024-bit MODP) and a pseudorandom function.
- SKEYSEED: the secret keys from which all keys are derived for IKE SA (SKe: encryption key to ensure confidentiality; SKa: authentication key to ensure integrity, and SKd: derivation key master secret to compute further CHILD SAs keys).
- IKE_SPI: stands for IKE Security Parameter Index. It uniquely identifies an IKE_SA.
- Lifetime: duration of an IKE SAs.
- Nonce: INITIATOR nonce (Ni) and RESPONDER nonce (Nr). These are randomly generated values to reinforce the security.
- Message ID counters: the ID counters provide anti-replay for IKEv2 exchanges by increasing the ID counter by one for every emitted IKEv2 message.
- IKEv2 window size: if the window size has a value of *N*, it implies that there can be *N* unacknowledged IKEv2 requests at any given time during communication.

During the Phase 2 of IKEv2 in Figure 6, the INITIATOR sends an IKE_AUTH request and the RESPONDER replies with an IKE_AUTH response.

When Phase 2 is finished, both nodes agree on the following parameters of their CHILD SAs:

- CHILD SA SPI: a 32 bits unique identifier of the CHILD SA.
- IP addresses: source/destination IP address of the IKEv2 compliant nodes.
- IPsec Protocol: AH or ESP.
- Sequence number counter: value to control every incoming/outgoing IP packet protected with IPsec, preventing replay or unauthorized re-injection of already processed IPsec traffic.
- Anti-replay window size *N*: any packet with the sequence number *X* + *N* is discarded, where *X* is the awaited sequence number.



Initiator | Responder

HDR, SK {IDi, AUTH, SAi2, TSi, TSr, N(INITIAL_CONTACT)} →

← HDR, SK {IDr, AUTH, Sar2, TSi, TSr}

HDR, SK {[N], SA, Ni, [KEi], [TSi, TSr]} →

← HDR, SK {SA, Nr [KEr], TSi, TSr}

Fig. 6. IKv2 Phase 2 exchange.

- ESP/AH information: encryption and/or authentication algorithms, keys, initialization values, and key lifetimes.
- Lifetime: time interval or byte count after which a SA must be replaced with a new SA (and new SPI).

Figure 7 shows the steps to maintain the SAs between two end points. Each SA is identified by using Security Parameter Index (SPI), destination address and AH or ESP. The SPI identifies the SA in the IPSec header. During the packet transmission or reception each sensor node holds a SAD as in steps 4 and 10. SAD will be used to get the information about SPI, keys, algorithms, etc. as shown in step 3 and 9. The node looks up the corresponding security association and fetches the necessary keys to apply security to the IP packet. For the initial negotiation between peers IKE uses the SPD to define how the data should be protected shown in step 5. Then, IKE can process the negotiation in step 7 to request for new associations.

4. Proposed architecture

We have proposed an adaptation of IKEv2 for HWSN titled as the Cooperative Key Exchange System (CKES) to prolong the network lifetime and energy efficiency. In CKES, each constraint node can request its neighbors (less constraint or unconstraint nodes) to process the heavy cryptographic operations. This system should maintain the same security policy to guarantee the confidentiality and data integrity.

As shown in Figure 8, CKES distributes the IPSEC_CHANNEL for constraint nodes and IKE_CHANNEL for less constraint or unconstraint nodes. The nodes will form a cluster and this will be coordinated with the help of a Cluster Head (CH) and a Highly Trusted Node (HTN) which has been chosen by the CH. For this scenario, the following assumptions have been considered:

- A Base Station (BS) will generate prime numbers mi where i varies from 1 to *n* (co-primes to each other).
- Sensor nodes will be organized into clusters based on LEACH clustering algorithm (Handy et al., 2002).

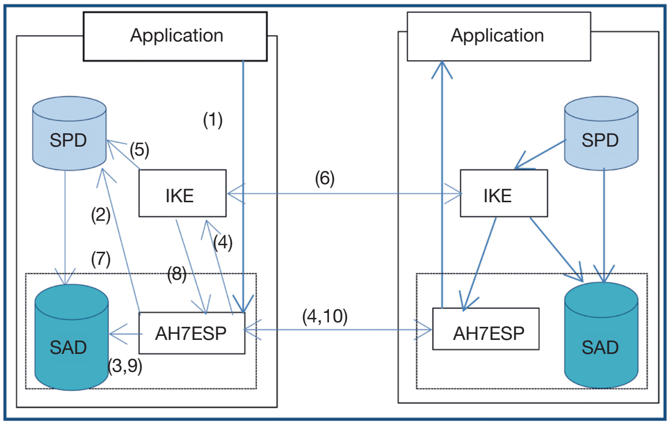


Fig. 7. IKE architecture.

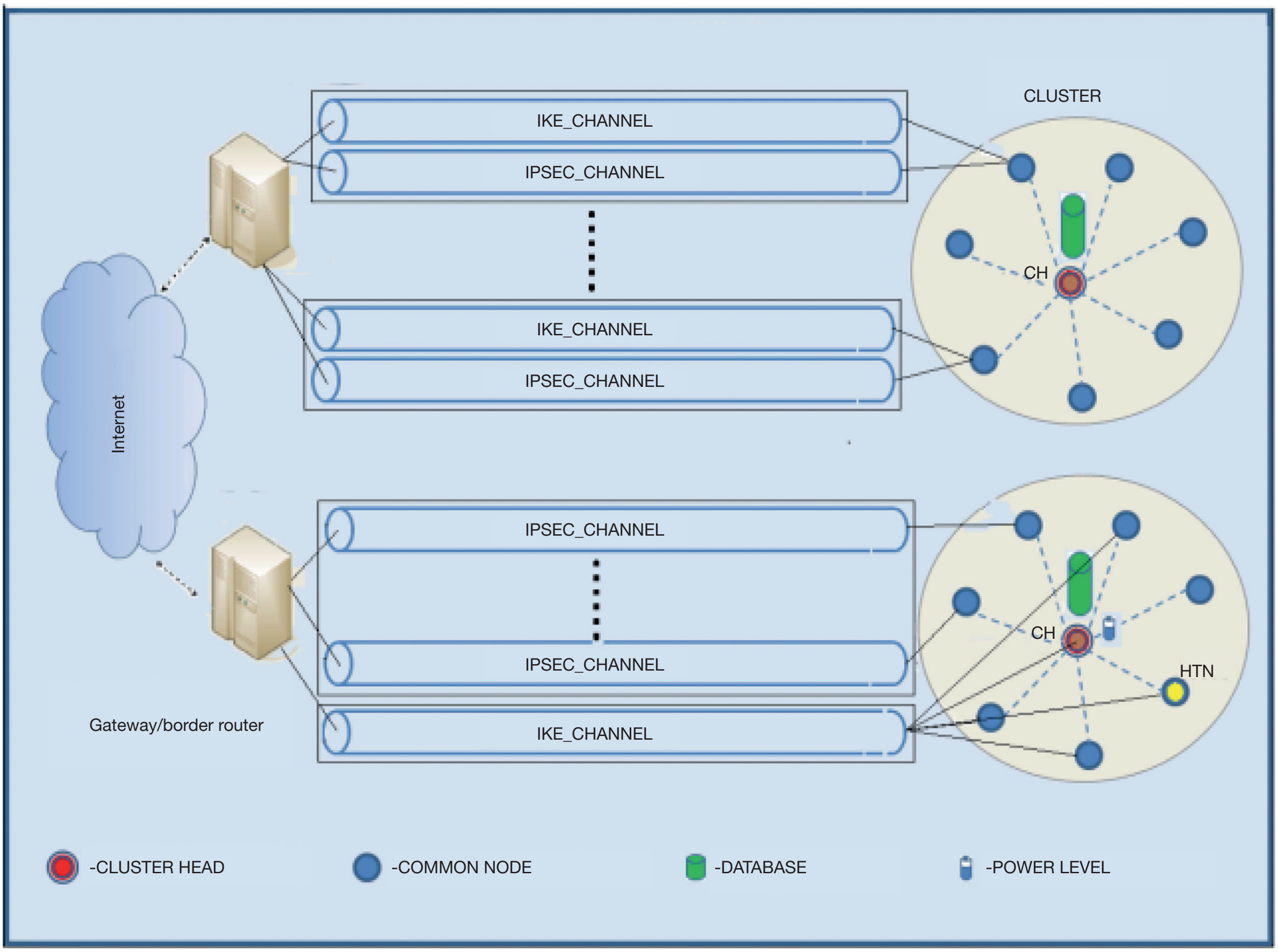


Fig. 8. Cooperative key exchange system (CKES) architecture.

- The less constraint and the unconstraint nodes will be assigned with the prime numbers generated by BS.
- Group-Based Trust Management Scheme (GTMS) will be used as a trust management system for Clustered Wireless Sensor Networks.
- The HTN has more resources in terms of battery power and memory capacity. This will also support IKEv2 negotiations and coordination between all collaborative nodes.
- The HTN can request CH to get the power level information of all nodes.

4.1. Heavyweight cryptographic operations

Most of the previous works on security of WSN were using signature scheme and the Diffie-Hellman (DH) key agreement protocol. In cryptography, this has been considered as the heavyweight algorithms. So the use of these algorithms will delay the communication, and need more power consumption and high resource utilization between the constraint nodes. As an experimental example on the MiCA2 platform (Meulenaer et al., 2008), the energy consumptions of the RSA-1024 signature and verification are respectively 359.87 and 12.04 mJ (Piotrowski et al., 2006). The memory space consumption to use the RSA signature is 4.4 kB of ROM (Read Only Memory) and 1 kB of RAM (Random Access Memory). The example of DH exchange given in Gaffari (2014), which consumes 1185 mJ to share the DH values and compute the master key.

In CKES, the heavyweight cryptographic algorithms mentioned in Table 2 will be moved from the constraint nodes to the less constraint or unconstraint neighbor nodes.

4.2. Definitions

In this subsection, the CRT and the DH key agreement algorithms have been detailed. The CRT algorithm is used for our proposed CKES and the DH key agreement protocol used during the IKE_INIT phase of IKEv2 protocol.

Table 2
Cryptography algorithms costs.

Operations	Energy consumption (mJ)	Operation time (s)
RSA_Sign	359.87	12.04
Sign_Verif	14.05	0.47
DH exchange	1185	54.11

4.2.1. Chinese Remainder Theorem

Let $m_1, m_2, m_3, \ldots, m_n$ be a set of prime numbers and it should be co-prime to each other. Let $a_1, a_2, a_3, \ldots, a_n$ be a set of positive integers such that $a_i < m_i \ \forall$ i ∈ [1..N]. Let S be a congruence system presented as in equation (4).

$$S = \begin{cases} x \equiv a_1 & (mod\ m_1) \\ x \equiv a_2 & (mod\ m_2) \\ & . \\ & . \\ & . \\ x \equiv a_k & (mod\ m_k) \end{cases} \tag{1}$$

CRT states that a unique solution for S exists and lies between [1, M – 1]. This unique solution is given by equation (5).

$$X = \sum_{i=1}^{k} (a_i * M_i * y_i)(mod\ M) \tag{2}$$

Where

$$M = m_1 *\ m_2 * \ldots *\ m_k \prod_{i=1}^{k} m_i \tag{3}$$

$$M_i = M / m_i \tag{4}$$

$$\text{and} \quad y_i = M_i^{-1}(mod\ m_i) \tag{5}$$

y_i is determined by using extended Euclid's theorem.

4.2.2. Diffie-Hellman (DH) key agreement protocol

DH key agreement algorithm (Rescorla, 1999) is a protocol which allows two peers A and B to do the key agreement for the safe communicate between them. After choosing a prime number 'p' and a generator 'g', A and B generate two secrets values a and b as shown in Figure 9. Then, they compute their public values $g^a\ mod\ p$ and $g^b\ mod\ p$ and exchange them at the last step.

4.2.3. Proposed CKES procedure

The procedure consists of eight steps as described below:

Phase 1: IKE_INIT_SA.

Step 1: A constraint node A (Initiator) requests HTN to start a key exchange session with a node B (Responder). In this request, A has to mention the B's ID and the maximum number of collaborative nodes.

Step 2: The HTN multicasts the request to N less constraint nodes which are available to support heavyweight cryptographic algorithms.

Step 3: Each requested Cluster Member (CM) starts to update its power level, computation power, availability and network threshold (Ct). Based on these values it accepts the CH request.

Step 4: The HTN sends 'k' collaborative CMs IDs to A and as well as their CRT coefficients (y_1*M_1,y_2*M_2, …, y_kM_k) given by the CH.

Step 5: A starts to generate secret value 'a' that will be used for the DH exchange and the master key computation. This value should be the sum of 'k' elements a_1, a_2, …, a_k such that $a_i < \min(\mathrm{m_i}) \ \forall \ \mathrm{i} \in [1..\mathrm{k}]$ and $a = \sum_{i=1}^{k} a_i$.

Step 6: A computes X using equation (5), but without applying modulo M. This X value generates a solution for CRT as in equation (4) and it satisfies a set of congruence $X = a_i\ (mod\ m_i)$, where i ∈ [1..k].

Step 7: A sends the solution X, the security association $\mathrm{SA_{i1}}$, and the Message Authentication Code MAC to the HTN.

Step 8: HTN multicast X to all collaborative nodes and sends the "IKE packet" which consists of (HDR, SAi1, Ni, CERT_HTN) to the responder B.

Step 9: After receiving X, each collaborative node computes its own, where $a_i\ = X\ mod\ (m_i)$ Then, it calculates the DH parties as $g^{a_i}\ mod\ p$ and sends it to the responder B.

Step 10: To compute the A's DH public key, B makes the product of the values received from the CMs as following:

$$\prod_{i=1}^{k} g^{a_i}\ mod\ p = g^{\sum_{i=1}^{k} a_i} mod\ p = g^a mod\ p$$

Step 11: After checking the certification and computing the master key $g^{a*b} mod\ p$, node B sends $g^b mod\ p$ to the HTN as well as the "IKE packet" (HDR, SAr1, etc.).

Step 12: Each collaborative node computes its own master key part $g^{b*a_i} mod\ p$ and sends it to the initiator A including the HTN part.

Step 13: After receiving the master key portions, A makes the product of received value in order to compute the Master Key

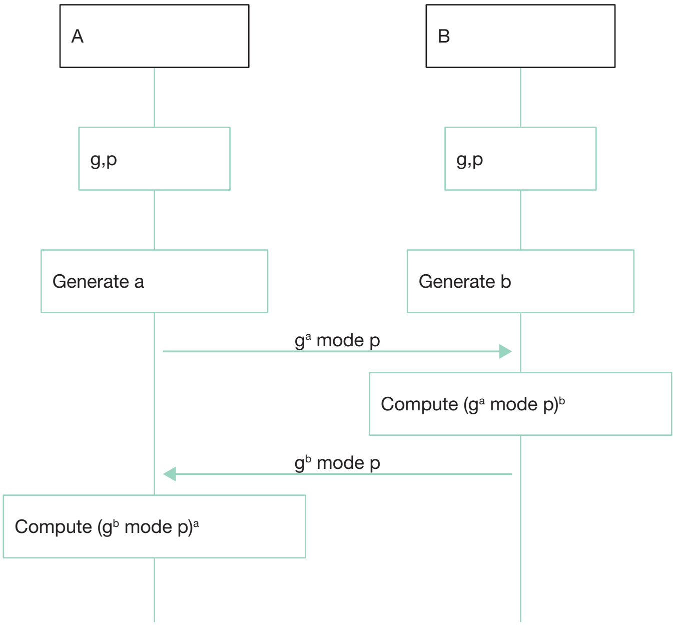


Fig. 9. Diffie-Hellman key agreement.

$$\prod_{i=1}^{k} g^{b*a_i}\, mod\ p = g^{b\sum_{i=1}^{k} a_i} mod\ p = g^{b*a} mod\ p$$

Step 14: Once the master key is calculated, peers A and B can start the second IKE exchange IKE_AUTH based on the negotiated SA and the master key.

All the details of CEKS protocol are illustrated in Figure 10.

5. Simulation

In this section, we present simulation results to evaluate the efficiency of the CKES compared with the basic IKEv2 implemented in (Kasraoui et al., 2014b). We carried out the simulations using NS2 simulator in which we have modified the energy model class to estimate the energy consumption of cryptography operations of each SN as well as the communication energy costs. This model presents a linear decrement in the computation of the residual energy.

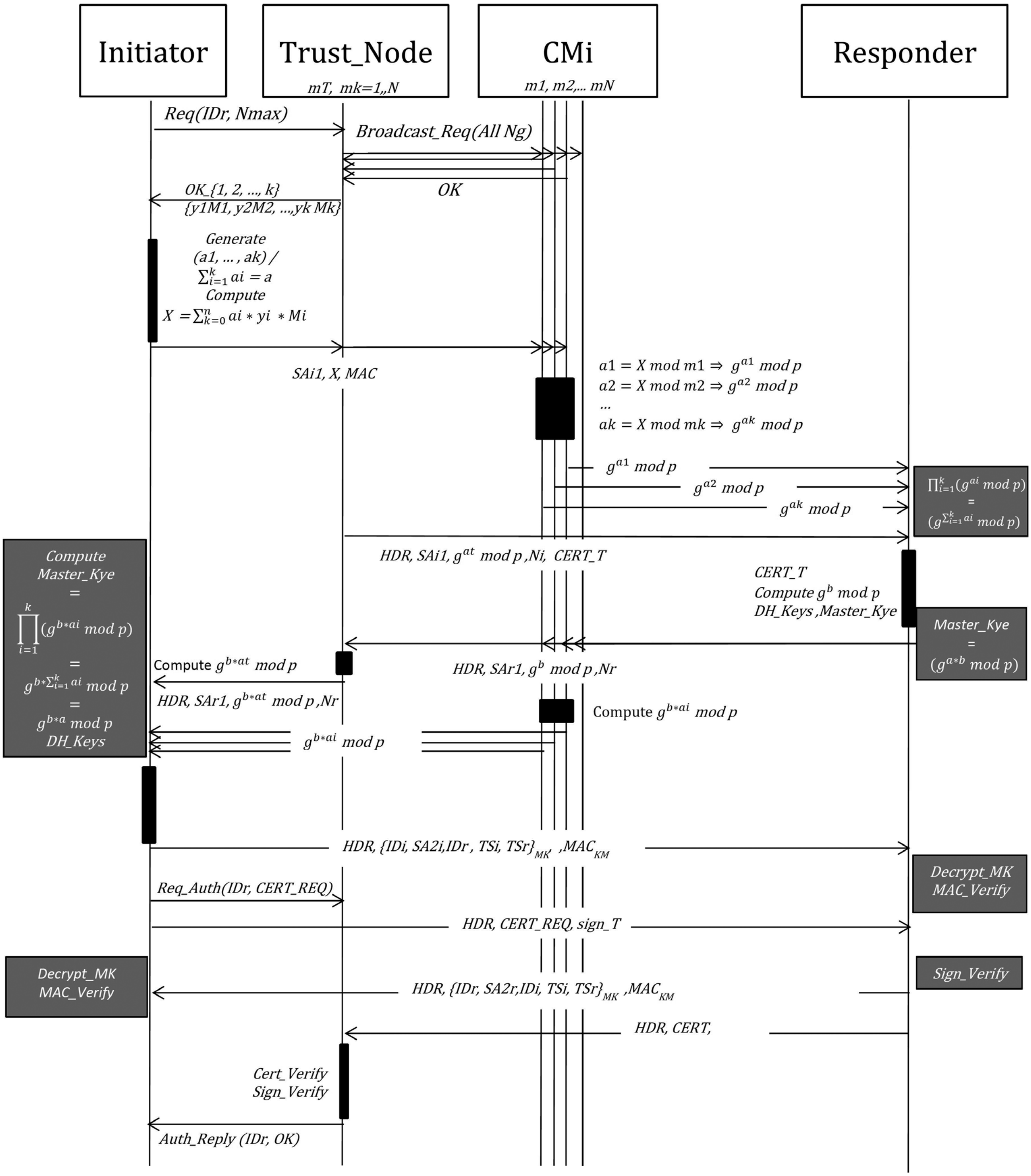


Fig. 10. Cooperative Key Exchange System in HWSN.

The communication energy costs depend on the data rate and transmission (Tx) and reception (Rx) power consumption during the Tx and the Rx. Taking the example, mentioned in the Table 3, of the energy model of the MICAz platform. This sensor node is based on the 8-bit ATmega128L microcontroller and running at 7.37 MHz (Eady, 2005). The power consumption in the transit mode is equal to 65 mW. If it works with 250 kbps data rate, then the node consumes 0,26 μJ to transmit 1 bit by using equation (9).

$$\text{Energy cost} = \left(\text{Tx power consummation}\left[W/s\right]\right) / \left(\text{data rate}\left[bps\right]\right) \quad \left[\text{joules}\right]\Big) \tag{6}$$

5.1. Simulation parameters

We have considered WSN in which a security gateway connects sensors to every other Internet host using IPsec protocol. We have used a network configuration of 80 wireless sensors and a single security gateway (SG). The simulation parameters are outlined in Table 4.

NS2 simulator is capable of producing performance measures at various protocol levels and observation points in WSNs. In our work, we have implemented the minimum IKEv2 features as described in RFC 5996 by using OpenSSL library for the cryptography operations:

- IKE_INIT_SA and IKE_AUTH phases for the initiator and responder nodes.
- DH protocol (groupe1).
- X.509 certificates, RSA signature.
- A simple traffic selector negotiation.
- One child SAs per IKE SA.
- AES-128 encryption algorithm and the SHA-1 hash function.

5.2. Simulation results

5.2.1. Communication costs

Figure 11 shows the communication energy cost of IKEv2 protocols implemented in NS2. The energy model is based on the reception and the transmission costs to compute the communication energy during the IKE_INIT and IKE_AUTH steps.

Table 5 shows the communication energy costs of IKEv2 and the proposed CKES and as well as the sent and received bytes.

The communication cost in the CKES is less than the IKEv2. In IKv2, the certifications and signatures verification or computation was done by the initiator. This consumes much energy. But in the proposed CKES, this computation will be happening in the collaborative nodes. This reduces the consumption of energy and the utilization of memory and this also improves the network lifetime by maintaining the same security level.

Table 3
Measured power consumption of the MICAz.

Power consumption	MICAz
Transmit	65 mW
Receive	72 mW

5.2.2. Energy consumption of cryptographic operations

Table 6 summarizes the total energy costs of both protocols. It shows that the most saved energy is related to the computation cryptography operations. Using CEKS, we can save around 90% of energy for the constraint nodes and we can also minimize the network energy consumption and maximize the network lifetime.

According to these results, our proposed cooperative approach is considered as a suitable key exchange system in HWSNs. In addition, we are studding the efficiency of our protocol at the network level and we aim to develop a framework based on CEKS, trust, and resource manager distributed approach.

Table 4
Simulation parameters.

Parameters	Value
Simulator	NS2 with Mannasim Patch
Traffic type	UDP
Bandwidth	250 kbps
Scenario size	400 m × 400 m
Transmission range of nodes	70 m
MAC protocol	IEEE 802.11
Routing protocol	AODV
Propagation model	Two ray ground
Simulation time	100 s
Initial energy	100 J

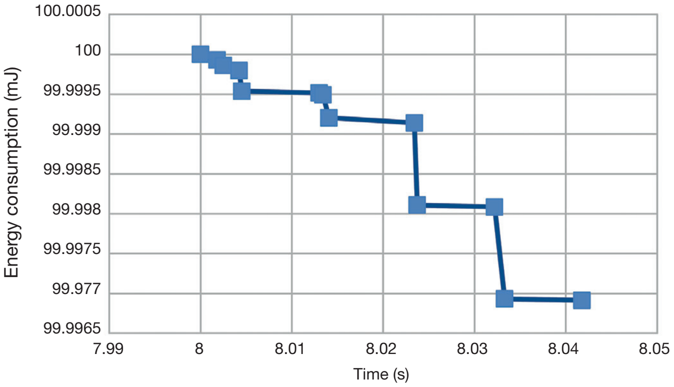


Fig. 11. Energy communication costs in IKEv2.

Table 5
Energy communication costs of the IKEv2 and CKES protocol.

IKEv2	Sent (bytes)	Recv (bytes)
IKE_INIT	124	124
IKE_AUTH	497	497
IKEv2 communication costs (μJ)	1291.68	1440.72
CEKS communication costs (μJ)	879.84	1357.2

Table 6
Energy computation costs of the IKEv2 and CKES protocol.

	IKEv2	CKES
Total computation costs (mJ)	252.87	4.8
Total communication costs (mJ)	3.08	2.73
Total costs (mJ)	255.95	7.53

6. Conclusions and future work

This paper has presented a CKES based on the concept of CRT. The proposed approach is an adaptation of the IKEv2 in IP based WSN. We have modified it in order to provide more balanced energy consumption and longer lifetime comparing to a basic IKEv2 implementation.

We have presented the details of the design and implementation of CKES in NS2. We have compared this with the IKE and implemented the main functionalities that can be used in WSNs. The improvement of key exchange system in WSNs which we have proposed with the help of this new module can offer a better lifetime of network.

Our future work would explore the possibility to add a trust management system which could play an important role to make decisions in the collaborative system. We also aim to develop a resources management system in order to improve the balance of energy consumption between SNs.

Finally, we aim to integrate our solution into the routing protocol already developed in Kasraoui et al. (2013).

Acknowledgements

This work has been supported by the Haute-Normandie regional council and the European institutions by the FEDER program

References

Casado, L., & Tsigas, P. (2009). A secure network layer for wireless sensor networks under the Contiki Operating System. *Lect. Note. Comput. Sci., 5838,* 133-147.

De Meulenaer, G., Gosset, F., Standaert, F., & Pereira, O. (2008). *On the energy cost of communication and cryptography in wireless sensor networks* (pp. 580-585). WiMob'08.

Dhurandher, S., Obaidat, M.S., Jain, G., Mani Ganesh, I., & Shashidhar, V. (2010). An efficient and secure routing protocol for wireless sensor networks using multicasting. *Proceedings of the IEEE/ACM International Conference on Green Computing and Communications, Green Com* (pp. 374-379). Hangzhou, China.

Eady, F. (2005). *Implementing 802.11 with Microcontrollers: Wireless Networking for Embedded Systems Designers.* Newnes.

Gupta, V., Millard, M., Fung, S., Zhu, Y., Gura, N., Eberle, H., & Shantz, S.C. (2005). Sizzle A standards-based end-to-end security architecture for the embedded internet (pp. 247-256). In *Pervasive Computing and Communications.*

Granjal, J., Silva, R., Monteiro, E., Silva J. S., & Boavida, F. (2008). Why is ipsec a viable option for wireless sensor networks (pp. 802-807). In *Proceedings of the 5th IEEE International Conference on Mobile Ad-Hoc and Sensor Systems (MASS '08).*

Granjal, J., Monteiro, E., & Silva, J.S. (2010a). A secure interconnection model for IPv6 enabled Wireless Sensor Networks (pp. 1-6). In *Proceedings of the IFIP Wireless Days (WD '10).*

Granjal, G., Monteiro, E., Sa Silva., J. (2010b). Enabling network layer security on ipv6 wireless sensor networks. In: *IEEE Global Communications Conference (GLOBECOM'10).* Miami, USA,

Gaffari, A. (2014). An energy efficient routing protocol for wireless sensor networks using A-star algorithm. *Journal of Applied Research and Technology, 12,* 815-822.

Handy, M.J., Haase, M., & Timmermann, D. (2002). Low energy adaptive clustering hierarchy with deterministic cluster-head selection (pp. 368-372). *4th International Workshop on Mobile and Wireless Communications Network.*

Hui, J., & Thubert, P. (2011). *Compression Format for IPv6 Datagrams over IEEE 802.15.4-Based Networks. RFC 6282.* Arch Rock Corporation, Cisco.

IEEE std. 802.15.4 (2003). *Wireless Medium Access Control (MAC) and Physical Layer (PHY) specifications for Low Rate Wireless Personal Area Networks (LR-WPANs).*

Jung, W., Hong, S., Ha, M., Kim, Y.J., & Kim, D. (2009). SSL-based lightweight security of IP-based wireless sensor networks (pp. 1112-1117). *Advanced Information Networking and Applications, Workshops.*

Karlof, C., Sastry, N., & Wagner, D.(2004). TinySec: A link layer security architecture for wireless sensor networks (pp. 162-175). In *Second ACM Conference on Embedded Networked Sensor Systems.*

Kent S., & Seo, K. (2005). *Security architecture for the internet protocol. RFC 4301.*

Kushalnagar, N., & Montenegro, G. (2007). *IPv6 over Low-Power Wireless Personal Area Networks (6LoWPANs): Overview, Assumptions, Problem Statement, and Goals. RFC 4919*

Kaufman, C., Hoffman, P., Nir, Y., & Eronen, P. (2010). *RFC 5996: Internet Key Exchange Protocol (IKEv2).*

Kasraoui, M., Cabani A., & Mouzna J. (2013). Zbr-M: A New Zigbee Routing Protocol. *International Journal of Computer Science and Applications, 10,* 15-32.

Kasraoui, M., Cabani, A., & Chafouk, H. (2014a). IKEv2 authentication exchange model in NS-2. In: *IEEE International Symposium on Computer, Consumer and Control.* Taiwan.

Kasraoui, M., Cabani, A., & Chafouk, H. (2014b). Formal verification of wireless sensor key exchange protocol using AVISPA. In: *IEEE International Symposium on Computer, Consumer and Control.* Taiwan.

Manral, V. (2007). *Cryptographic algorithm implementation requirements for encapsulating security payload (ESP) and authentication header (AH). RFC 4835.*

Montenegro, G., Kushalnagar, N., Hui, J., & Culler, D. (2007). *RFC 4944 - Transmission of IPv6 Packets over IEEE 802.15.4 Networks.*

Piotrowski, K., Langendoerfer, P., & Peter, S. (2006). How public key cryptography influences wireless sensor node lifetime (pp. 169-176). *Proceedings of the fourth ACM workshop on Security of ad hoc and sensor networks.* New York, NY, USA.

Rescorla, E. (1999). *Diffie-Hellman Key Agreement Method. RFC 2631*

Raza, S., Chung, A., Yazar, D., Voigt, T., & Roedig, U. (2011). Securing communication in 6LOWPAN with compressed IPsec. In: *7th International Conference on Distributed Computing in Sensor Systems.* Barcelona, Spain.

Raza, S., Duquennoy, S., Höglund, J., Roedig, U., & Voigt, T. (2012). Secure communication for the internet of things — A comparison of link-layer security and IPsec for 6LoWPAN. In: *Security and Communication Networks.* New York: Wiley.

Yu, H., He, J., Zhang, T., Xiao P., & Zhang, Y. (2013). Enabling end-to-end secure communication between wireless sensor networks and the Internet. *World Wide Web Journal, 16,* 515-540.