Configurable Link Layer Security Architecture for Wireless Sensor Networks

Devesh C. Jinwala, Dhiren R. Patel and Kankar S.Dasgupta

***Abstract*—Ensuring network security in Wireless Sensor Networks (WSNs) indeed is critical. Due to the data-centric multi-hop communication in WSNs, an essential consideration in the security solutions for WSNs is to ensure security features at the link layer. The link layer security can be implemented in hardware or in software. The existing software based link layer security architectures do not offer configurable security.**

In this paper, we propose a novel design of link layer security architecture for WSNs. The principal characteristics of the design we propose, is the flexible and configurable architecture, with respect to the actual security attributes demanded by the application. Our design is based on the premise that when the link layer architecture is implemented in software, flexibility and seamless integration of the application code become the prime advantages. We also emphasize that with the increasing computational, storage and bandwidth resources of the sensor nodes, we can get good performance and efficiency from software implementation of link layer architecture also.

Index Terms— **Wireless Sensor Networks, Cryptography, Network Security.**

I. Introduction

Wireless Sensor Networks (WSNs), comprise of the networked wireless sensor nodes to realize some vital functionality. Some of the example applications of WSN are environmental monitoring applications (flood, water level, temperature, stress, strain, pressure etc), industrial automation in hostile environments, tracking movements of living beings in parks, sanctuaries, offices, schools, banks etc, surveillance in war zones, enemy camps and a host of others. No matter what the applications are, the security of the network nodes by themselves and that of the data collected and disseminated; is of prime concern.

Wireless sensor nodes are characterized by severe constraints in power, computational resources, memory, bandwidth and small physical size with low power consumption [1]. The security protocols used for conventional networks demand higher computational, storage and energy resources. Therefore, ensuring security in WSNs requires adapting the conventional security protocols to the resource constrained WSN environment. Also, the WSNs need to support security at the link layer also, apart from the same at the application layer [4].

We also re-emphasize the view that unconditional security is not possible in practice – neither in conventional networks nor in resource constrained environment. Therefore, the security demands of an application should be derived from the *operational paradigm* followed and the *execution environment* used.

The existing research attempts at providing software-based link layer security solutions in WSNs are SPINS [5], TinySec [4], SenSec [6], MiniSec [7]. These architectures assume abstract applications and abstract security models of the WSN deployment. Not all the applications require *all* the security attributes of confidentiality, message authentication, replay protection etc. Hence, the link layer security architecture used must allow configurable security i.e. tuning the desired security parameters and tuning the level of the security with respect to the application. We enlist various security parameters that a link layer design should consider in section III.

In this paper, therefore, our *principal contribution* is the design proposal of an alternate - security attributes driven - link layer security architecture. The security architecture mentioned can be feasibly implemented, to integrate with any suitable mote operating system like TinyOS [8]. The IEEE 802.15.4 [9] standard compliant RF transceiver chip viz. CC2420/CC2430/CC2431 [10] offer on-the-fly encryption, message authentication and replay protection.

We believe that with the improvement in the computational, storage and bandwidth resources of the motes, the security framework now can feasible be implemented in software, too. So, the innovative approach in our solution will offer technological benefits:

- configurable and flexible security features which can be tailored to the needs of the application under consideration
- serving as an experimental test-bed for security related experimentations easily e.g. if one desires to test a new cipher designed, it can easily be done by merely *plugging-out* the default cipher and *plugging-in* the new cipher.
- seamless migration of legacy applications to make them security enabled. There is no separate API is required to be designed or implemented to use the security features. Therefore, when enabling security in any existing

Manuscript received April 12, 2008. This work was supported in part by the Ministry of Human Resource Department, Govt of India Grants.

Devesh C. Jinwala is with the National Institute of Technology, Ichchhanath, Surat, Gujarat, India, 395007. Phone: 91-261-2259765, fax: 91-261-2228394; e-mail: dcjinwala@gmail.com).

Dhiren R. Patel is with the National Institute of Technology, Ichchhanath, Surat, Gujarat, India, 395007. Phone: 91-261-2201595, fax: 91-261-2228394; e-mail: dhiren29p@gmail.com)

Kankar S. Dasgupta is with the Space Applications Centre, Indian Space Research Organization, Ambavadi, Ahmedabad (e-mail: ksdasgupta@yahoo.com)

applications, minimal changes to the existing application code is to be made to make the applications *security-enabled*.

The design is unique in the sense that none of the existing link layer architectures, integrating into the sensor node operating framework use such approach – although similar approach is advocated in the IEEE 802.15.4 standard.

The proposed security architecture is initially intended to be employed in the TinyOS [8] environment, but can be extended for any other alternate platform. The design proposed is backed up by our experimentations on the TinyOS platform.

The rest of the paper is organized as follows: in section II we discuss the generic design features of link layer architecture, in section III focuses on our design of the proposed architecture, in section IV we compare our work with the related work, whereas we conclude with the plan for the further work in section V.

II. DESIGN OF A LINK LAYER SECURITY ARCHITECTURE

The conventional security protocols are based on *route-centric* multi-hop communication, wherein the intermediate nodes are not required to inspect the data content of the packets. Whereas, the WSNs exhibit the *data-centric* multi-hop communication paradigm wherein, the intermediate nodes carry out some form of data processing (aggregation, summarization, duplicate elimination) on the incoming packets data packet to be routed towards the base station. The advantage of such data processing is the reduction of the overall communication costs. Such on the fly processing of the data is called *in-network* processing. [2][3]

Because of in-network processing, the end-to-end communication paradigm, exhibited by the commonly used applications on the Internet alone, is not suitable for the WSNs [4]. Instead, appropriate link layer security architecture is required to obtain the desired security attributes at each wireless link.

We now propose the criteria for the design of a link layer security architecture:

1. Security Properties: The link layer security architecture must offer the required security attributes viz. (a) data encryption yielding confidentiality of data (b) associating either an un-keyed message digest OR a symmetric cipher based keyed message authentication code (MAC) yielding message integrity (c) replay protection yielding conservation of precious energy resources by non-acceptance of the old but replayed data packets to the destination (d) freshness (strong freshness/weak freshness) (e) support for the state-of-the-art cipher with attack resistant key-size.

2. Performance (a) the link layer framework must offer efficient operation with low computational, storage and energy usage overhead. Typically, according to the authors in [4], the performance of link layer security architecture is satisfactory if the resource overhead with security enabled is within 10% of the overhead without security enabled.

3. Flexibly auto-configurable – (a) with respect to the available processor power, available memory, the radio chip used (bandwidth/data rate) (b) with respect to the application demands supporting either of only data authentication and data encryption; OR data authentication and replay protection; OR data encryption, data authentication and replay protection and OR data secrecy, data authentication, replay protection and freshness.

The security goals above are achieved using the proven techniques of (a) employing cryptographic transformations using a block cipher (b) associating message authentication code (MAC) using a block cipher so that only those recipient nodes which share the symmetric key with the sending node can verify the authenticity of data (c) associating a message digest using a hashing algorithm so that any entity can check the hash code to verify the authenticity of data (d) using nonce, counters, time-stamps to associate some identity with the packet, so as to distinguish between a replayed packet and genuinely transmitted.

The size of the MAC associated with the packets is an important design criterion. Typically, for a WSN with the maximum band-with of 19.2 kbps, a MAC of 4-bytes is suitable. Because as in [4], adversary will need to make 232 sustained attempts without being detected, to forge the MAC. With 64-bytes packet size, it will require a more than 20 months by an adversary to do so. But if the maximum bandwidth is 250 kbps, forging the 4-byte MAC would require merely 90 days of continuous attempts to do so (again assuming 64-bytes packet size). In the WSNs left unattended for months, such an attack is possible. Hence, we need higher bytes for the MAC. In short, depending upon the frequency of the data transmitted and maximum bandwidth permitted, configuration of security features should be permissible.

The usage of the security enabling techniques does indeed involve increase in the length of the control information associated with the packet, increased computational power, increased energy consumption and greater demands on storage. Therefore, the emphasis should be on ensuring tolerable performance degradation to achieve the additional benefits due to security.

III. OUR DESIGN AND ANALYSIS

We propose to modify the TinyOS operating framework with the augmentation of the secure communications stack. Our design is unique is in the fact we propose the secure communication stack to be configurable depending upon the data rate, level of security desired and the nature of the application (i.e. the security attributes desired).

Our architecture is aimed to support different modes of operations as shown in Table I.

As can be observed from the table, nine different modes of operations serve to tune the security attributes in line with those actually demanded by the application under consideration. The motivation for such a configurable architecture with different modes is [12]. With the help of security-attributes driven taxonomy of typical WSN applications, it is shown in [12] that different WSN applications demand different levels and types of security.

For example, for the typical environmental or habitat monitoring applications like (a) tracking the movement of an animal in a sanctuary or (b) monitoring the amount of rainfall in the catchments areas of a river across a dam, to enable forecasting the probability of rainfall downstream; the confidentiality of data packets transmitted en route to the

base station, is not essential. But the data integrity, entity authentication, message freshness and replay protection are very vital for the same applications. Also, for the two applications under consideration, the frequency of packets transmission also would also vary, thereby demanding different bandwidth

On the other hand, in security-sensitive applications in military, health and socioeconomic domains like tracking the movement of an enemy troop OR monitoring the health parameters of a patient OR human security systems; it is essential to ensure confidentiality of the data packets transmitted along with some or all of the other attributes listed above.

The frequency of transmission of the data packets and the associated radio bandwidth has a very significant bearing on the number of bytes employed for the MAC e.g. in a typical Smart Office application of the WSN wherein a WSN is deployed in a Smart Kindergarten for monitoring the behavior of the children and their movements with video streaming [17]; the volume of the data transferred as well as the frequency at which the data is transferred is high. S if a mere 4-byte MAC is used, the probability of a MAC being forged by an adversary operating with in, is definitely higher than if an 8-byte MAC were employed. Hence, we propose variable MAC sizes which the application designer using our communication architecture, can choose, configure and implement in the application deployed; thereby optimizing the level and type of security.

The third vital component of our design is the support for replay protection. The layer, at which the support for replay protection ought to be there in WSNs, has well been debated in the literature. In [4], the authors advocate that the support for replay protection should be taken care off, by the application layer itself. As compared to that, in [7], the designers of MiniSec include the same as a vital component of their link layer framework.

Following the strategy that *any void left in the security framework, with the intention of the same being filled-in by the implementers of the application* can be a serious security threat; we offer the option of replay protection for our framework.

Next, we elaborate the design features with a discussion of the modes of operations offered:

The first option is named as Null wherein it is assumed that because the security support is implemented in the hardware, the proposed security framework in the operating system does not include any security feature. This option can be used while employing the security enabled radio chips for the WSNs.

The second one viz. FlexiSecHASH is proposed to offer the support only for message authentication – typically suited for the applications demanding only message integrity; without any demands for the same being checked only by the designated nodes. Hence, we follow the un-keyed authentication technique i.e. hashing employing an algorithm like SHA1. Any participating entity can check the authenticity of the message, irrespective of the keys employed or not.

FlexiSecAUTH64 mode, the third mode, is intended to be employed for High Volume data and high data rate applications like PODS at Hawaii [18] OR SSIM application [19] for Process control applications involving monitoring of machine parameters. Here, we intended to provide the support for 64 bits of MAC but without data encryption; because we believe confidentiality of data is not demanded here. This mode is suitable for the high data rate environmental monitoring applications, of the kind mentioned above.

Similarly, the fourth option viz. FlexiSecAUTH32 offers *authentication-only* support for low data rate applications typically found in environmental control e.g. water-level monitoring, flood forecasting, stress monitoring in concrete structures etc.. In such applications, it is sufficient to sense and transmit only a few packets per day with only minimal parameter values.

The fifth and the sixth options viz. FlexiSecAUTH_ENC64 and FlexiSecAUTH_ENC32 are the options to support data confidentiality apart from the message integrity using 8-byte MAC and 4-byte MAC respectively. These modes are useful in the applications requiring confidentiality e.g. in the military applications of the kind referenced before.

Again, here a unique feature of our design is the selection of the Output Codebook Mode (OCB) [20] mode which ensures encryption and generation as well verification of message authentication code (MAC – known as Message Tags in OCB) in a single pass. As compared to CBC mode used for authentication in contemporary link layer architectures, using OCB mode will definitely save computational resources and thereby energy resources too.

Indeed in the preliminary experimentations carried out by us OCB mode has been observed to be conserving significant amount of storage over the corresponding Cipher Block Chaining (CBC) mode [21]. We expect the same savings exhibited for energy consumption also – firstly due to the savings in computational and storage usage and secondarily due to the single iteration carried out to achieved dual functionality of confidentiality and message authentication.

Note that the same mode for block cipher modes cannot be used if the application demands only message authentication since this mode does not support keyed authentication alone.

The seventh and the eight modes in our proposed architecture are the FlexiSecAUTH_REPP64 and FlexiSecAUTH_REPP32 modes. These modes augment the security properties attained in the previous modes with that of replay protection. These modes could be employed for applications demanding message authentication as well as replay protection alone, without any encryption. Therefore, these modes are intended to employ CBC MAC [22] and CBC mode as the message authentication scheme and block cipher mode of operation respectively.

The last mode of operation viz. FlexiSec_ AUTH_ENC_REPP64 basically is intended to offer all the security attributes listed above with a message TAG MAC of 8 bytes using the OCB mode again – could be employed in highly security-critical applications.

Thus, the proposed architecture indeed offers flexibly configurable security attributes for the applications.

We here do not specify a particular block cipher for the proposed security architecture. The design indeed is

attempted to be clear of any dependence on a specific cipher. But, in view of the increasing trend towards more powerful sensor nodes (with increased storage resources, increased computational power and higher battery life) and an 128-bit key size for the block ciphers, we intend to implement an optimized version of the block cipher AES (Rijndael) in its 128/128/10 rounds configuration [16].

IV. RELATED WORK

Link layer security may be implemented either in hardware or in software. The solutions in hardware certainly offer better performance as compared to the ones implemented in software.

An example of such a solution implemented in hardware is the security solution embedded in the transceiver chips for the sensor nodes e.g. IEEE 802.15.4 [9] standard compliant RF transceiver chip from Texas Instruments/Chipcon viz. CC2420/CC2430/CC2431 [10], designed for low-power and low-voltage wireless applications with the support for data encryption, data authentication amongst others like packet handling, data buffering, burst transmissions, clear channel assessment, link quality indication and packet timing information. But, the solutions in hardware lack the flexibility that one often needs in experimenting with the security solutions and can not remain configurable, with respect to the actual security demands of the application.

There have been significant efforts to design and implement the link layer security architecture in software.

Some of the notable ones are the TinySec [4], SenSec[6] and MiniSec[7].

TinySec proposed in [4] designed for the Berkeley Mica Motes in 2004, is commonly accepted standard link layer security architecture immensely popular as a test bed for evaluating many security related WSN protocols and applications. The performance overhead with security enabled in TinySec, is within the 10% of the case where the packets are sent without security features being incorporated.

The principle characteristics of TinySec are (a) a light weight and efficient link layer security package (b) Skipjack as the block cipher with only 80-bit key size (c) integrated into the TinyOS 1.1 operating system to enable developers easily integrate into sensor network applications (d) a research platform that is easily extensible and has been incorporated into higher level protocols.

However, (a) it does not support all of the link layer security goals mentioned above viz. 1c, 1d, 1e, 3a and part of 3b. (b) the four-byte MAC is suitable for radios operating at 19.2 kbps but not suitable for contemporary radios operating at 250 kbps (g) it is devised only for Crossbow's Mica, Mica2, Mica2dot motes and does not support may other popular motes including Crossbow's Iris motes[23], Intel's Imote [24] and Moteiv's Telos motes.

Tieyan Li et al, propose an alternate link layer security architecture viz. SenSec in [6]. SenSec draws upon its basic design from TinySec but offers encryption as well as authentication by default. SenSec was designed principally to suit a specific application viz. Automatic body monitoring for patients and environmental monitoring for habitat. The principle characteristics of SenSec are: (a) designed specifically for specific application - not meant to act as a research platform (b) uses Skipjack as the block cipher with only 80-bit key size but with modified modes. But, SenSec (a) is devised only for Mica, Mica2, Mica2dot motes (d) does not support the link layer security goals viz. 1c,1d, 1e and part of 3.

Luk Mark et al propose another alternative architecture viz. MiniSec, designed for the Telos motes [11], in [7]. MiniSec follows a different approach in that it offers two operating modes, one tailored for single source communication, whereas the other, for multi-source broadcast communication. It offers all the basic desired link layer security properties viz. data encryption, message integrity and replay protection. The characteristics of MiniSec are (a) link layer security architecture devised for the Telos motes (b) Skipjack as the block cipher with only 80-bit key size (c) low energy consumption and high Security (d) devised for IEEE 802.15.4 compliant radio chip CC2420. But, MiniSec (a) employs a MAC of 4 bytes only – with a transceiver bandwidth of 250 kbps and adversary continuously attempting to forge the MAC, forging the MAC is possible (f) it does not support the security goal viz. 3 above i.e. the security support is not configurable. In fact, MiniSec does not even support authentication-only (i.e. without the support for encryption) mode of crypto operations. As per the survey in [12], a large percentage of the existing WSN applications in socio-environmental setup demand only data authentication and NOT data encryption. MiniSec would not be suitable in such environment.

ZigBee [13] is a popular IEEE 802.15.4 compliant specification for a suite of high level communication protocols using small, low-power digital radios. It is targeted at RF applications that require a low data rate, long battery life and secure networking. ZigBee provides a higher level of security as compared to that in TinySec; but at the same time, it exhibits higher communication overhead and high energy consumption by the radio. Also, the use of ZigBee protocol involves appropriate licensing and membership of the ZigBee Consortium.

IEEE 802.15.4 specification suffers from the fundamental design limitations, in that (a) it offers an option of using security without authentication and (b) it uses the COUNTER mode for confidentiality - which is labeled as a security loophole [14]. In fact, Sastry et. al. in [14], give a detailed analysis of the security loopholes in IEEE 802.15.4 specifications.

Nevertheless, as pointed out earlier, none of the solutions above, offer link layer security that is tunable with respect to the security demands of a wide range of applications – to serve as an efficient, easy-to-work-with and globally acceptable experimental platform for link layer security based research in WSNs.

V. CONCLUSION AND FUTURE WORK

The security-attributes driven link layer security architecture is highly desired for the WSN applications. We propose a lucid requirement analysis and basic design of such architecture. We again emphasize that to the best of our knowledge, this is first published literature presenting the design of configurable link layer architecture.

We further intend to fully simulate the architecture using

the TOSSIM [25] simulator and actually test the implementation using a variety of motes.

VI. Acknowledgment

We thank the anonymous reviewers who took pains and spared their valuable time for giving the critical comments, in order to make this paper take the shape it has taken, now.

References

[1] I F Akyildiz, W Su, Y Sankarasubramaniam, E Cayirci; "Wireless Sensor Networks: A Survey", *Computer Networks*, 38(4), March 2002

[2] V Raghunathan, C Schurgers, Park S, Srivastava M B; "Energy Aware Wireless Microsensor Networks"; *IEEE Signal Processing Magazine*, Vol 19, Issue 2, March 2002.

[3] Adrain Perrig, John Stankovic, David Wagner; "Security in Wirless Sensor Networks"; *Communications of the ACM, CACM'04*, Vol 47, No 6, 2004

[4] Chris Karlof, Naveen Sastry, David Wagner; "TinySec: Link Layer Encryption for Tiny Devices", *ACM Conference on Embedded Networked Sensor Systems*, 2004, Ohio.

[5] A Perrig, R Szewczyk, V Wen, D Cullar, J D Tygar; "SPINS: Security Protocols for Sensor Networks"; *Proceedings of the 7th International Conference on Mobile Computing and Networking*, July 2001

[6] Tieyan Li, Hongjun Wu, Xinkai Wang, Feng Bao; "SenSec Design, I^2R Sensor Network Flagship Project"; Technical Report TR v1.0

[7] Mark Luk, GhitaMezzour, Adrian Perrig, Virgil Gligor; "MiniSec: A Secure Sensor Network Communication Architecture"; *ACM International Conference on Information Processing in Sensor Networks*; April 2007

[8] Jason Hill, Robert Szewczyk, Alec Woo, Seth Hollar, David Culler, Kristofer Pister; "System Architecture Directions for Networked Sensors"; *ASPLOS*, 2000

[9] The IEEE 802.15.4 - IEEE Standard for Information technology, Telecommunications and information, exchange between systems, Local and metropolitan area networks Specific requirements IEEE Computer Society, September 2006

[10] Radio Transciever Chips CC2420/CC2430, CC2431 – http://www.ti.com

[11] Telos motes – http://www.moteiv.com

[12] Devesh Jinwala, Dhiren Patel, K S Dasgupta; "A Security Attributes driven taxonomy of Wireless Sensor Network Applications"; *International Conference on Sensors and Related Networks (SENET 07);* sponsored by VIT, Indian Nuclear Society and University of Applied Sciences, Germany; at Vellore Institute of Technology (VIT), Vellore, Dec 2007

[13] ZigBee Alliance. ZigBee specification. Technical Report Document 053474r06, Version 1.0, ZigBee Alliance, June 2005.

[14] Sastry N and Wagner D; "Security considerations for IEEE 802.15.4 networks"**;** *Proceedings of the 3rd ACM workshop on Wireless security;* 2004

[15] M. B. Srivastava, R. R. Muntz, and M. Potkonjak, "Smart kindergarten: sensor-based wireless networks for smart developmental problem-solving environments," *Proceedings of the 7th ACM International Conference on Mobile Computing and Networking*, Italy, 2001

[16] Announcing the Advanced Encryption Standard; Federal Information Processing Standards Publication, November 26, 2001.

[17] Levis P, Lee N, Welsh M, Culler D; "TOSSIM: accurate and scalable simulation of entire TinyOS applications"; *Proceedings of the 1st international conference on Embedded networked sensor systems*; 2003

[18] PODS, A Remote Ecological Micro-sensor Network Project, http://www.pods.hawaii.edu

[19] L. Schwiebert, S. K. S. Gupta, and J. Weinmann, "Research challenges in wireless networks of biomedical sensors"; *Proceedings of the 7th ACM International Conference on Mobile Computing and Networking (MobiCom '01)*, 2001.

[20] Phillip Rogaway, Mihir Bellare, John Black, "OCB: A block-cipher mode of operation for efficient authenticated encryption", *ACM Transactions on Information and System Security (TISSEC)*, Volume 6, Issue 3, pp.365-403, August 2003

[21] M. Bellare, A. Desai, E. Jokipii, and P. Rogaway, "A concrete security treatment of symmetric encryption: Analysis of the DES modes of operation" *Proceedings of 38th Annual Symposium on Foundations of Computer Science (FOCS 97)*, 1997.

[22] Mihir Bellare, Joe Kilian, Phillip Rogaway, "The security of the cipher block chaining message authentication code", *Journal of Computer and System Sciences*, Vol 61 Isssue 3, pp.:362-399, December 2000.

[23] IRIS motes - http://www.xbow.com/Products/wproductsoverview.aspx

[24] Intel's Imote - http://www.xbow.com/Products

[25] P Levis , N Lee, M Welsh, D Culler ; TOSSIM: accurate and scalable simulation of entire TinyOS applications; Proceedings of the 1st international conference on Embedded networked sensor systems; 2003

TABLE I

Sr No	Name	Description
1	Null	Security support in hardware radio chip
2	FlexiSecHASH	Naïve Authentication Support with one-way hash function SHA1
3	FlexiSecAUTH64	64 bits - 8 bytes – MAC : only keyed authentication – CBCMAC
4	FlexiSecAUTH32	32 bits - 4 bytes – MAC : only keyed authentication – CBCMAC
5	FlexiSecAUTH_ENC64	8 bytes MAC and encryption – OCB (single pass)
6	FlexiSecAUTH_ENC32	4 bytes MAC and encryption – OCB (single pass)
7	FlexiSecAUTH_REPP64	8 bytes MAC: keyed authentication (CBCMAC) & replay protection
8	FlexiSecAUTH_REPP32	4 bytes MAC: keyed authentication (CBCMAC) & replay protection
9	FlexiSec_ AUTH_ENC_REPP64	8 bytes MAC: keyed authentication (OCB), encryption & replay protection