Trusted Wireless Sensor Node Platform

Yusnani Mohd Yussoff, Habibah Hashim, *IEEE*

Abstract— ***Security problems related to Wireless Sensor Networks (WSN) has directly influenced the credibility of WSN applications and its services. With the advancement and demand in the WSN applications such as in military, structural health monitoring, transportation, agriculture, smart home and many more, the system stands to be exposed to too many potential threats. Currently, WSN systems has mainly relied on software based security for protection against breach of systems and data integrity. As WSN systems become more ubiquitous, software based security is regarded as no more sufficient for WSN applications. This paper discusses security issues in WSN area and reviews work done on hardware base security. Based on the preliminary works that have been carried out, Trusted Platform Module (TPM) initiatives by Trusted Computing Groups (TCG) together with Trusted Zone technologies by ARM are seen as possible approaches toward better security implementation in WSN. Finally, this paper proposes new embedded security implementation utilizing 32-bit ARM11 processor with trustzone features to enhance the integrity of the sensor node platform.***

Index Terms—**Wireless sensor network, Trusted Computing, Trustzone, security**

I. Introduction

Consideration for security in Wireless Sensor Networks (WSN) may arise from four different perspectives. The most important aspect is the limited resources in sensor node that includes CPU limitation, memory constraint and restricted energy supply. Seconds, is the nature of wireless communication that is open to anyone to hear or to intercept. Moreover, the lack of fixed infrastructure also imposes security concerns where malicious nodes are able to join the networks in an untraceable manner. Finally, WSN are also highly exposed to the risk of physical attacks due to sensor nodes being normally unattended. These issues are briefly discussed in [1]. In addition, WSNs share the same security threats as those which exist in other communication networks which include message interception, modification and fabrication as well as interruption of communications and operations. Kuorihelto in his paper concludes that the limited capabilities in WSN nodes in fact leads to higher and more effective attack from attackers using advanced tools [2]. These are some of the work that mentioned the importance of security in WSN and constraints faced by these networks.

Research in the security area of WSN covers development of new security algorithms that consume low energy and memory [3-6], comparison of energy efficient security algorithm including Public Key Cryptography (PKC) and symmetry cryptography technique [7-10] and finally hardware implementation of security algorithms [11-17].

In sensor nodes, energy-efficient security can be established through software and hardware implementation. Software implementation refers to security algorithm installed in the memory and used when needed while hardware implementation refers to security protocol hardwired into a processor or extra security chip on the sensor node. This can be viewed as in Fig. 1.

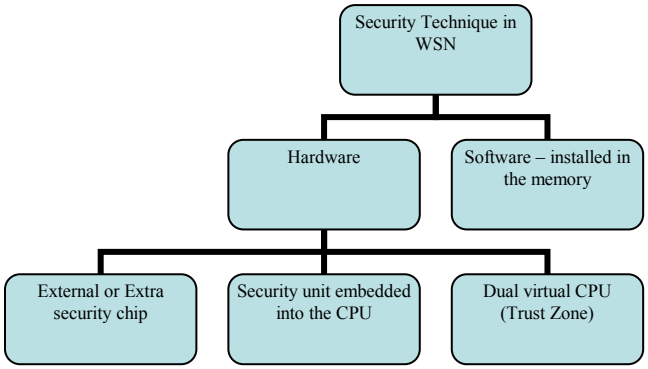

Fig.1: Security Implementation Technique in WSNs.

In software implementation, researchers look for simplified algorithms that offer similar or higher security level but overcome constraints in the sensor node. While a good number of research are focused on developing the most suitable cryptography algorithm for sensor node , this paper will concentrate only on the hardware based security architectures in the sensor node. This work is basically prompted from the study that shows better performance of sensor network security for hardware implementation.

The rest of the paper is organized as follows: Section II present security challenge in WSN area. Section III briefly discuss the trusted platform techniques followed by section IV which focuses on the related studies on hardware base security for WSN and subsequently section V present the proposed security work. Finally section VI concludes the paper.

II. Security Challenge

Majority of the work done in WSN security currently focuses on the security of the network without considering data privacy and the authenticity and integrity of the wireless sensor network nodes[18]. Future applications such as medical health, military, system monitoring, smart home and many more, demand higher security level that include access control, explicit omission or freshness, confidentiality, authenticity and integrity [19]. Detailed analysis of security demand in various type of applications with potential security threats can be found in [9]. Fig. 2, briefly shows the security

Manuscript received February 10, 2010. This work was supported in part by the Faculty of Electrical Engineering, Universiti Teknologi MARA,Malaysia and Ministry of Higher Education Malaysia.

Y.M Yussoff is currently a PhD student with the Faculty of Electrical Engineering, Universiti Teknologi MARA, Malaysia. (phone: 603-5543-5094;fax:603-5543-5577;e-mail:yusna233@salam.uitm.edu.my)

H.Hashim is with Faculty of Electrical Engineering, Universiti Teknologi MARA,Malaysia. (e-mail: habib350@salam.uitm.edu.my).

ISBN: 978-988-17012-9-9
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

goal of WSN based on F.Amin [9] paper and N.Verma thesis [19]. In order to achieve the above goals, Public key Cryptography (PKC) is believed capable to support asymmetric key management as well as authenticity and integrity. Although the use of PKC in WSN is previously denied due to its high resourced (energy, memory and computational)[20, 21], many recent work have proved its feasibility in the WSN area [1, 3, 6, 9-11, 15, 17, 22, 23]. Latest, Wen Hu [18] used Trusted Platform Module [24] hardware which is based on Public Key (PK) platform to augment the security of the sensor node. Their work claim that the SecFleck architecture provides internet level PK services with reasonable energy consumption and financial overhead.

It can be concluded that the demand for higher security levels in WSN increase significantly with the advancements in WSN applications. As mentioned earlier, the feasibility of PKC in WSN security is proven and therefore the choice of PKC as the best cryptography protocol in WSN area is unquestionable. The concern now is what is the best method to implement PKC in the sensor node and is it secure to run security protocol in on unsecured platform considering the nature of the WSN node that is normally expose to software attack and physical attack? Security provided by cryptography depends on safeguarding of cryptographic keys from adversaries. Therefore there is a need to adequately protect the keys to ensure confidentiality and integrity of sensitive data.

Fig. 1 describes the taxonomy of security implementation in sensor node or embedded system in general. At this stage, the authors believe that embedding security unit in the processor is the most suitable technique for securing wireless sensor node. This technique is believed to be capable of reducing the size of the sensor node, decreasing the processing time and preventing software attacks as well as providing other benefits. Johann et.al in his paper [25] also conclude that hardware based security features need to be integrated into the processor to avoid vulnerabilities such as those which exist in today's personal computer.

Besides secure implementation, the node also should communicate in a trusted environment. Tiago and Don [26] mentioned that the demand in trusted computing is driven by the potentially severe economic consequences due to unsecured embedded applications.

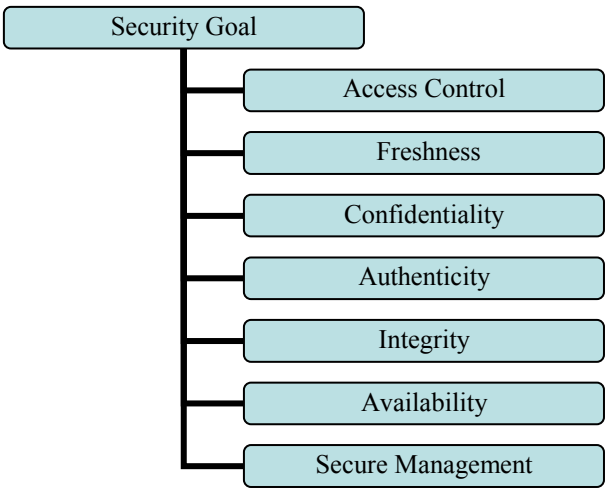

Fig. 2: Common Security Goal

III. Trusted Platform Technique

It is believed that nothing is secured and can be trusted. With enough time and money, attackers will definitely find a way to break and attack any systems. Therefore a clear definition of a trusted system is needed. According to [27], trust can be define as an entity that always behaves in the expected way for the intended function. Basic properties of a trusted computer or systems can be listed as below.

- *Isolation of programs* – prevent program A from accessing data of program B
- *Clear separation between user and supervisor process* – there should be a systems to prevent user applications from interfering with the operating system.
- *Long term protected storage* – secret values are stored in a place that last across power cycles and other events.
- *Identification of current configuration* – provide identity of the platform and software or hardware executing on it.
- *Verifiable report of the platform identity and current configuration* – a way for other users to validate a platform.
- *Hardware basis for the protections* - protection is a combination of hardware and software.

Demand on a trusted platform in the network environment arrived when merely software based mechanisms became inadequate to provide desired security level. Trusted Computing Platform Alliance (TCPA) [28] was formed in late 90's and finally emerged as trusted computing group (TCG) in 2003[29]. TCG's basically worked to develop an inexpensive chip that helps users protect their sensitive information. Muhammad Amin et.al [30] in his paper discussed on trends and directions in trusted computing. His paper provides details on advancement of trusted hardware to facilitate security that led to the design and implementation of TCG specific solution. This paper also claims that ARM is the only trusted implementation available for secure embedded applications.

The following section discusses two alternatives that can be used to establish trusted and secure security systems followed by review on hardware-based security implementation.

A. Trusted Platform Module

Trusted Computing Groups (TCG) [24] solves security problems through operating environments, applications and secure hardware changes to the personal computer. TCG used secure hardware Trusted Platform Module (TPM) chip as a basis for trusted computing that provides a level of relevant since hardware based security is difficult to compromise than conventional approaches.

TPM verifies the integrity of the system through trusted boot, strong process isolation and remote attestation that verify the authenticity of the platform. Encryption and decryption used RSA algorithm with default 2048-bit, SHA-1 hash and random key generator. TPM can be implemented in a dedicated chip, co-processor or software based [31] where the connection of TPM is vendor specific and is not specified by TCG [28]. Fig. 3 briefly shows block diagram of TPM consisting of ten components to accelerate security processes.

Unfortunately, the choice of RSA and SHA-1 algorithms has made the platform unsuitable for WSN applications. RSA

ISBN: 978-988-17012-9-9
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

with 2048 bits has been confirm to consume higher energy and therefore unsuitable for WSN applications and embedded system [9]. Moreover, RSA when implemented in hardware demand large silicon area and therefore increase the size of the chip [17]. An alternative to RSA is Elliptic Curve Cryptography (ECC) and Advance Encryption System (AES). Beside RSA, the choice of SHA-1 is also mooted. Recent research indicates that many cryptographers doubt the security of SHA-1 and recommend against the used in new design.

To conclude, TPM model is not the best choice for secure or trusted platform implementation in embedded system especially in WSN applications due to the performance and security concern. Most importantly, the TPM is designed for the personal computer which does not usually have concerns on resource constraints.

B. Trust Zone in ARM Microprocessor

The key feature of ARM trust zone is "secure to the core". The security features are hard wired into the microprocessor core and therefore promise an extra degree of security over a software only approach and extra security chip [32].

ARM trust zone is specifically designed for smart phones, handheld devices and embedded systems that can potentially be compromised by malicious hackers. The nature of WSN that exposes it to too many types of attacks and intrusions demand extra security features that not only support security but also trustworthiness.

Wilson et. al [33] in his paper viewed trustzone in ARM as a dual-virtual CPU Systems. The running software looks at the trustzone as two separate virtual processors. The virtualization is achieved through hardware extension within the CPU design. The extensions annotate whether the core is running normal world or Secure World Software and propagate these selections to the memory and peripherals. With this implementation, the secure memory and peripherals can reject the non-secure transactions.

The switching between secure and non-secure world in the ARM processor is established through Secure Monitor Call (SMC) instruction and interrupts. In line with WSN constraints, the trust zone in ARM processor eliminates the need for extra security chip. Moreover, security elements can be executed at full processor speed without cache-flushing overhead. Beside it can also save the power as only one of the two virtual processors run at one time. Fig. 4 shows how trustzone mimics two processors.

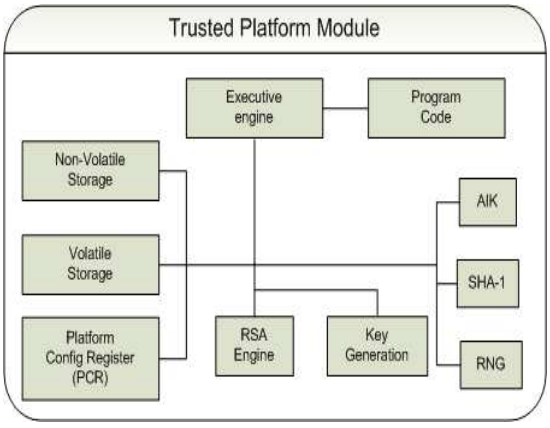

Fig. 3: Standard TPM Components

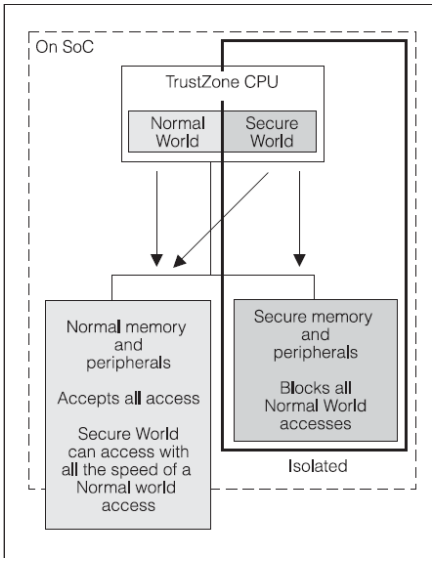

Fig. 4: One core support two operating worlds: secureworld and normal world. Cortesy of: Wilson.P et.al [33]

IV. RELATED STUDIES

G.Edward Suh et.al [34] in his work presented an AEGIS secure processor architecture that secure the embedded system beyond normal security algorithm. AEGIS, a single-chip secure processor, introduces mechanisms that not only authenticate the platform and software but also protect the integrity and privacy of applications from physical attacks. Two new techniques are introduced to overcome physical and software attacks in WSN, PUFs and off-chip memory.

Physical Random Function (PUFs) is a function that generates secret number so that users can authenticate the processor that they are interacting with. With PUFs the secret are generated dynamically by the processor and therefore provide higher physical security compared to storing the secrets in non-volatile memory. Besides, PUFs also do not need any special manufacturing process or special programming and testing steps.

Off-chip memory mechanisms ensure the integrity and the privacy of off-chip memory by encrypting and decrypting all off-chip memory data transfer using a one-time pad encryption scheme. To summarize, AEGIS can protect embedded devices from any attacks before program execution, during the execution and also from physical and software attacks through the security mechanism designed. Unfortunately, the added hardware mechanisms had increased the size of the processor core and marginally degrade program performance.

Lie et. al. [35] from Stanford University introduced Execute Only Memory (XOM) that enabled copy and tamper resistant software distribution to prevent software piracy. All data leaving the machine is encrypted using symmetric-key encryption and the keys are specifically distributed to each processor using public-private key pair. This technique provides a software tamper-resistant execution environment that is established through tagging or encryption. Unfortunately, hardware assist is considered necessary in XOM architecture to provide fast symmetric ciphers.

SecFleck [18] which was mentioned earlier used external TPM chip on the sensor node. This TPM based public key platform facilitates message security services with

ISBN: 978-988-17012-9-9
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

confidentiality, authenticity and integrity. SecFleck platform consists of hardware and software module and later connects to the Fleck sensor node board. Although the evaluation on the computation time, energy consumption, memory footprint and cost is reasonable and positive, the extra platform connected to the sensor node is unacceptable for sensor node applications. Beside the security algorithm used is not aligned with sensor node constraints.

Another work on hardware based security is done by [36, 37] where both works developed a co-processor for security algorithm. While the first work developed RSA co-processor, the second work implements an AES co-processor (VHDL design only) for resource constraint embedded system. RSA co-processor was implemented on Altera Stratix FPGA development board. Both works claim to have better speed and area compared to other research and commercial implementation.

Latest, two studies have embarked on the development of trusted and secure platform utilizing ARM11 trustzone architecture. Johannes Winter[38] and Xu Yang-ling[39], both utilize Linux kernel 2.6 and ARM trustzone features. While Johannes merge trustzone features with TCG-style trusted computing concepts, Mobile Trusted Module (MTM), Xu integrate the Mandatory Access Control (MAC) in Linux kernel 2.6 with the trustzone features to enhance the security up to the non-secure environment. The first has designed a robust and portable virtualization framework for handling non-secure guest and the second work presented an embedded system security solution.

V. PROPOSED WORKED

This work proposes the development of a sensor node platform utilizing ARM11, a 32-bit processor. This work was prompted due to lack of highly secured sensor node platform to accommodate future wireless sensor networks applications. While almost all available sensor node platforms [40] utilize software based security, this work proposed the use of trustzone feature in the ARM11 processor to enhance the security level by limiting the security parameter to a single chip. All important keys and data will be saved in the On-SoC memory thus preventing the platform from shack and lab attack. Basically the sensor node platform will consist of ARM11 chip, external memory Flash and SDRAM, Zigbee as transmitter, temperature sensor and battery operated power supply.

A. Security Architecture

The primary goals are to assert the integrity of the software images executed in the sensor node platform by preventing any unauthorized or malicious modified software form running. Beside the design will ensure the confidentiality and integrity of the data during communications.

The above objectives are established through proper security architecture designed utilizing ARM trustzone features.

- Secure world – all the sensitive resources will be placed in the secure world memory locations. Refer to Fig. 5 for details of memory architecture. Trustzone Address space controller (TZASC) is used to configure regions as secure or non-secure. All non-secure transaction will be rejected to a region that is configured as secure. This ensures the confidentiality of important data.
- Single physical core – safe and efficient execution of code from both normal and secure world. This allows high performance security software to run alongside with normal world operating environment. Secure monitor code will be developed to switch from normal to secure and vice versa.
- Secure boot – Running secure boot algorithm to ensure the integrity of the programmes and devices on the platform.
- On-Soc RAM, ROM and crypto accelerator will ensure no highly sensitive data leaves the chip thus eliminating the possibility of shack and lab attack.

By using ARM trustzone, a small on-chip security system is presented in Fig. 5 below to execute the above objectives. It clearly depicts the permanent secure place and dynamic secure place that are accessible through AXI2APB bus system which has the capability to switch from secure process and non-secure process. Trustzone Memory Adapter (TZMA) will secure a region within an on-SoC memory such as SRAM where the secure location will be in the lower part of the memory region.

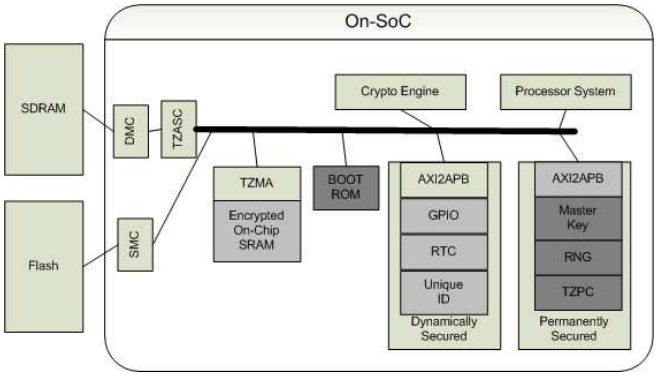

Fig. 5: Proposed security architecture for sensor node using ARM11 with Trust zone features.

Trustzone Address Space Controller (TZASC) will reject any non-secure transaction to a region that is configured as secure. Therefore external memory also can be partitioned into secure and non-secure region. Compared to previous works, the proposed security architecture has extended the security infrastructure throughout the system design. Almost all possible or potential loop holes have been blocked by embedding the memories, cryptographic accelerator, Random Number Generator and Master key into the processor chip. Instead of protecting assets in a dedicated hardware block, this architecture has made the valuable assets secured in the most protected location.

On top of the hardware design, a suitable security protocols such as secure boot will also be configured to complete the security design. Secure boot with code located in On-SoC ROM will provide a sequence with chain of trust for all the secure world software and hardware peripherals and some of the normal world software. Due to limited space, the secure boot flow will be discussed in another paper. With secure boot, the integrity of the OS image, software and peripherals on the platform can be verified to be truly unadulterated.

ISBN: 978-988-17012-9-9
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

Communications right after the secure boot process can be confirmed coming from a trusted sensor node.

Table I clearly depicts the advantage of the proposed security mechanism over previous work. Although the security level of the second technique (referring to Table I) is comparable with the proposed work, this offers proposed scheme extra advantages in term of power consumption and overall performance. While in AEGIS for example two processors are needed to run secure and normal process, in trustzone the dual virtual CPU will execute one of the processes (secure or non-secure) at one time thus eliminate extra processing work and reducing the chip size. Finally, since extra chip on the embedded applications board are not desirable, the first technique or work can be considered as irrelevant for WSN security implementation.

VI. CONCLUSION

Trusted Platform Module (TPM) implementation and other previous security implementation are comparable in terms of security with the proposed work. Two dominant features that differentiate this work from others are the location of sensitive resources such as the crypto keys and the denial of extra or dedicated processor core for security purposes. This implementation ensures no sensitive resources leaves the chip and therefore block most types of attacks. Besides that it also saves the silicon area and power consumption and also allows high performance security software to run alongside with the normal world operating environment. It is hoped that the outcome from this work can contribute towards higher security level in the area of WSN. Finally the choice of ARM11 as the main processor for the sensor node is in line with the constraint faced in sensor node development as it is rated as the most efficient processor in MIPS/Watt [41].

REFERENCES

[1] R. Wang, “Secure and Effiecient use of Public Key Cryptography in Sensor Networks,” Computer Engineering, Syracuse University, 2007.

[2] M. Kuorilehto, M. Kohvakka, J. Suhonen *et al.*, *Ultra-Low Energy Wireless Sensor Networs in Practice*: John Wiley & Sons Ltd., 2007.

[3] K. Woo Kwon, L. Hwaseong, K. Yong Ho *et al.*, "Implementation and Analysis of New Lightweight Cryptographic Algorithm Suitable for Wireless Sensor Networks." pp. 73-76.

[4] A. Perrig, R. Szewczyk, J. D. Tygar *et al.*, “SPINS: security protocols for sensor networks,” *Wirel. Netw.,* vol. 8, no. 5, pp. 521-534, 2002.

[5] E. O. Blab, and M. Zitterbart, "Towards Acceptable Public-Key Encryption in Sensor Networks." pp. 88-93.

[6] Y. oren, and M. Feldofer, "A Low-Resource Public Key Identification Scheme for RFID Tags and Sensor Nodes."

[7] M. R. Doomun, and K. Soyjaudah, “Analytical Comparison of CryptographicTechniques for Resource-Constrained Wireless Security,” *International Journal of Network Security,* vol. 9, no. 1, pp. 82-94, 2009.

[8] M. Feldhofer, and C. Rechberger, “A Case Againts Currently Used hash Functions in RFID Protocols,” 2006.

[9] F. Amin, A. H. Jahangir, and H. Rasifard, "Analysis of Public-Key Cryptography for Wireless Sensor Networks Security."

[10] A. S. K. Pathan, and H. Choong Seon, "Feasibility of PKC in resource-constrained wireless sensor networks." pp. 13-20.

[11] L. Huai, X. Zou, Z. liu *et al.*, "An Energy Efficient AES-CCM Implementation for IEEE802.15.4 Wireless Sensor Networks." pp. 394-397.

[12] G. E. Suh, C. W. O'Donnell, and S. Devadas, “AEGIS: A single-chip secure processor,” *Information Security Technical Report,* vol. 10, no. 2, pp. 63-73, 2005.

[13] G. Gaubatz, J.-P. Kaps, E. Ozturk *et al.*, "State of the ART in Ultra Low Power Public key Cryptography for Wireless Sensor Network."

[14] V. Ekanayake, I. Clinton Kelly, and R. Manohar, “An ultra low-power processor for sensor networks,” in Proceedings of the 11th international conference on Architectural support for programming languages and operating systems, Boston, MA, USA, 2004.

[15] Y. K. Lee, k. Sakiyama, L. Batina *et al.*, "Elliptic-Curve-Based Security processor for RFID."

[16] A. L. Huang, and W. T. Penzhorn, "Cryptographic Hash Functions and Low-Power Techniques for Embedded Hardware." pp. 1789-1794.

[17] O. Kocabas, E. Sabas, and J. Grobschadl, "Enhancing an Embedded Processor Core with a Cryptographic Unit for Performance and Security ". pp. 409-414.

[18] W. Hu, P. Corke, W. C. Shih *et al.*, "SecFleck: A public key technology platform for wireless sensor networks," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics).* pp. 296-311.

[19] N. Verma, “Practical Implementation and Performance Analysis On Security of Sensor Networks ” Full Thesis, Department of Computer Engineering, Rochester Institute of Technology, Rochester, new York, 2006.

[20] Y. Wang, G. Attebury, and B. Ramamurthy, "A Survey of Security Issues in Wireless Sensor Networks."

[21] W. Yong, G. Attebury, and B. Ramamurthy, “A survey of security issues in wireless sensor networks,” *Communications Surveys & Tutorials, IEEE,* vol. 8, no. 2, pp. 2-23, 2006.

[22] N. Gura, A. Patel, A. Wander *et al.*, *Comparing Elliptic Curve Cryptography and RSA on 8-Bit CPU*: Spingler Berlin, 2004.

[23] G. Gaubatz, J.-P. Kaps, and B. Sunar, "Public Key Cryptography in Sensor networks-Revisited*," *Security in Ad Hoc and Sensor networks*, pp. 2-18: Springer-Verlag, 2005.

[24] T. C. Groups. "TRusted Platform Module(TPM) Summary," 8 July 2009, 2009; http://www.trustedcomputinggroup.org/resources/trusted_platform_module_tpm_summary.

[25] J. Grobschadl, T. Vejda, and D. Page, "Reassassing the TCG Specifications for Trusted Computing in Mobile Embedded Systems." pp. 84-90.

[26] T. Alves, D. Felton, and ARM, “TrustZone: Integrated Hardware and Software Security,” *Technology In-Depth,* vol. 3, no. 4, pp. 18-24, 2004.

[27] D. Grawrock, *Dynamics of a Trusted Platform*: Intel Press, 2009.

[28] D. Grawrock, "Trusted Computing Platform Alliance," 2009.

[29] "Trusted Computing Group," www.trustedcomputinggroup.org.

[30] M. Amin, S. Khan, T. Ali *et al.*, "Trends and Directions in trusted Computing: Models, Architectures and Technologies."

[31] J. Grobschadl, T. Vejda, and D. Page, “Reassessing the TCG Specifications for Trusted Computing in Mobile and Embedded Systems,” 2008.

[32] T. R. Halfhill, "ARM DONS ARMOR: Trustzone Security Extensions Strengthnen ARMv6 Architecture," *MIcroprocessor*, [4 Nov 2009, 2003].

[33] P. Wilson, A. Frey, T. Mihm *et al.*, “Implementing Embedded Security on Dual-Virtual-CPU Systems,” *IEEE Design and Test of Computers,* vol. 24, no. 6, pp. 582-591, 2007.

[34] G. E. Suh, C. W. O'Donnell, and S. Devadas, “Aegis: A Single-Chip Secure Processor,” *IEEE Des. Test,* vol. 24, no. 6, pp. 570-580, 2007.

[35] D. Lie, C. Thekkath, M. Mitchell *et al.*, “Architectural support for copy and tamper resistant software,” *SIGPLAN Not.,* vol. 35, no. 11, pp. 168-177, 2000.

[36] A. PANIANDI, “A Hardware Implementation of Rivest-Shamir-Adleman Co-Processor for Resource Constrainted Embedded Systems,” Faculty of Electrical Engineering, Universii Teknologi Malaysia, Skudai, 2006.

[37] L. Y. Pin, “Verilog Design of a 256-bits AES Crypto Processor Core,” Faculty of Electrical Engineering, Universiti Teknologi Malaysia, Skudai, 2009.

ISBN: 978-988-17012-9-9
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

[38] J. Winter, "Trusted computing building blocks for embedded linux-based ARM trustzone platforms," in Proceedings of the 3rd ACM workshop on Scalable trusted computing, Alexandria, Virginia, USA, 2008.
[39] Y.-l. Xu, W. Pan, and X.-g. Zhang, "Design and Implementation of Secure Embedded Systems Based on Trustzone." pp. 136-141.
[40] M. Healy, T. Newe, and E. Lewis, "Wireless Sensor Node hardware: A review." pp. 621-624.
[41] M. A. M. Vieira, C. N. Coelho, Jr., D. C. da Silva, Jr. *et al.*, "Survey on wireless sensor network devices." pp. 537-544 vol.1.

Table I: Comparison Study on Trusted Implementation for Wireless Sensor Network

Technique	Definition	Advantage	Disadvantage
1. External hardware (TPM, AES and RSA chip)	Inclusion of a dedicated hardware security module outside of the main processor	Separate chip. Allows high levels of tamper resistance and physical security.	Sensitive resources leave the chip. Increase area and power consumption
2. Internal hardware (AEGIS, XOM, SP-Processor)	Hardware security modules that is located within the SoC.	Significant cost reduction performance improvement over external hardware. Security is comparable to trust zone technique.	Restricted perimeter and only capable of securing the cryptographic key material
3. ARM11 with Trustzone (proposed work)	Hardware architecture that extends the security infrastructure throughout the system design. Trustzone architecture enables any part of the system to be made secure.	Significant cost reduction Performance improvement over external h/ware. Only one process exist at one time (secure or non-secure)- reduce power Secure all sensitive resources.	

ISBN: 978-988-17012-9-9
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)