21

Integrity Enhancement in Wireless Sensor Networks

Yusnani Mohd Yussoff, Husna Zainol Abidin and Habibah Hashim
Faculty of Electrical Engineering, Universiti Teknologi MARA, Malaysia

1. Introduction

Consideration for security level in Wireless Sensor Networks (WSN) should depend on the demand of the intended applications. As energy consumption increase linearly with security level, the security designer should carefully choose the best security technique and the most suitable security parameters enough to protect the intended application. With the advancement and demand of WSNs applications in areas such as the military, structural health monitoring, transportation, agriculture, smart home and many more, the system stands to be exposed to too many potential threats. It is generally considered that applications such as smart home, transportation and agriculture need no security or be less secure compared to military and medical applications. However, sensor networks make large-scale attacks become trivial when private information on the entire system can instantly reach the hand of attackers. Due to the nature of WSNs that are left unattended and limited resources, there exist an urgent need for higher security features in sensor nodes and its overall systems. Without it, attackers with their own intentions and targets combined with their capabilities and sophisticated tools will always become a threat to future WSNs applications. However, latest technology in embedded security combined (low power, on-SOC memory, small size) with trusted computing specifications (ensuring trusted communication and user) is believed to enhance security features for future WSNs applications.

To this instant, research in the security area of WSNs covers development of new security algorithms that consume low energy and memory (Perrig *et al.*, 2002), comparison of energy efficient security algorithm including Public Key Cryptography (PKC) and symmetry cryptography technique (Pathan & Choong Seon, 2008) and finally hardware implementation of security algorithms (Ekanayake *et al.*, 2004, Gaubatz *et al.*, 2005, Huai *et al.*, 2009, Huang & Penzhorn, 2005, Kocabas *et al.*, 2008a, Lee *et al.*, 2008, Suh *et al.*, 2005). Our work is basically inspired by (Grobschadl *et al.*, 2008) suggesting hybrid implementations in securing WSNs applications.

The rest of the paper is organized as follows: Section 2 presents security challenges in WSN area. Section 3 briefly define physical attacks in WSNs. Section 4 will discusses the trusted

platform techniques followed by section 5 which focusses on the related studies on hardware based security for WSN and subsequently section 6 presents the proposed security work. Finally section 7 concludes the paper.

2. Security Challenges in WSN

Security challenges in WSNs can be divided into three different categories that are related to each other. [1] *Network*–Ensuring reliable, secure and trusted communication. [2] *Data*–Ensuring the integrity of the transmitted and processed data and finally [3]*Platform*-Guarantee the integrity of the sensor node exist in the network. Future applications such as medical health, military, system monitoring, smart home and many more, demand higher security levels that include access control, explicit omission or freshness, confidentiality, authenticity and integrity (Verma, 2006). Detailed analysis of security demand in various types of applications with potential security threats can be found in (Amin *et al.*, 2008a). Fig. 1, briefly shows common security goals of WSN based on the works of F.Amin and N.Verma . In order to achieve the above goals, PKC is believed to be capable of supporting 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) (Yong *et al.*, 2006), many recent works have proved its feasibility in the WSN area (Kocabas *et al.*, 2008b). Latest, Wen Hu (Hu *et al.*, 2009) used Trusted Platform Module hardware which is based on Public Key (PK) platform to augment the security of the sensor node. They claim that the SecFleck architecture provides internet level PK services with reasonable energy consumption and financial overhead.

Future applications such as medical health, military, system monitoring, smart home and many more, demand higher security levels that include access control, freshness, confidentiality, authenticity and integrity (Verma, 2006). Detailed analysis of security demand in various types of applications with potential security threats can be found in (Amin *et al.*, 2008a). Listed goals in Fig. 1, are achievable through PKC implementation supporting 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) (Yong *et al.*, 2006), many recent works have proved its feasibility in the WSN area (Kocabas *et al.*, 2008b). Latest, Wen Hu (Hu *et al.*, 2009) used Trusted Platform Module hardware which is based on Public Key (PK) platform to augment the security of the sensor node. They 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 has been established. 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. While majority of the work

done in WSN security have focused on the security of the network (Hu *et al.*, 2009), our proposed works will consider the three challenges describe earlier to secure the WSNs applications from software and physical types of attacks. Beside we will also ensure smallest security parameter in our overall security design.

At this stage, the authors believe that embedding the security parameters 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 and physical attacks as well as providing other benefits. Johann et al. in his paper (Grobschadl *et al.*, 2008) 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 (Alves *et al.*, 2004) mentioned that the demand in trusted computing is driven by the potentially severe economic consequences due to unsecured embedded applications. Following section will only consider security design for the third type of security challenges with the intention to secure the sensor node from physical attacks and ensure the integrity of the sensor node in the network.

3. Physical Attacks in WSN

Effect on attacks to WSNs applications can either be direct or indirect. While the first can cause disclosure of private information, modification and falsification of data and sensor node failure, the latter will basically cause unreliable services to the WSNs applications such as low data rate, service breakdown and inconsistent communication. Both effects are mostly the result of physical attacks or node tampering.

Tampering
Tampering as defined by A.Becher et.al (Becher *et al.*, 2006) is the ability to get full access to the node and it involves a modification to the internal structure of the chip. Physical attacks on the other hand are referring to attacks that require direct physical access to the sensor node. W.Znaidi et al. On the other hand, defined tampering as an action that involved physical access and node capture (Znaidi *et al.*, 2008). To avoid terminology problem, 'tampering' in this paper is as defined by A.Becher et al. and is seen as impossible in WSNs application as it involved sophisticated tools and takes a longer time to complete (Base station may have terminated communication with this sensor node by this time). Therefore it is not as likely to happen as the attacks that can be carried out in the field.

Physical Attacks
As defined earlier, physical attacks refer to attacks that involves direct connection with the sensor node. Adversaries may perform the attack by connecting their sophisticated tools on the site or taking away the sensor node. Their intention might vary from just to destroy the sensor node to extracting private information to be authenticated or authorized in the network. Sensor nodes can usually be attacked through the JTAG port that is widely used during the development phase and for debugging. With the JTAG port being enabled, adversaries will have the capability to take control of the whole system. Another form of attack is by exploiting the Bootstrap Loader (BSL) and this mostly happens during the boot up

process. With having access to the boot devices and debug session, attackers will be able to study the systems and its operation thus providing them with enough information to clone the system, insert malware and disturb the overall operations of the sensor node and its systems.

Although a total solution to physical attacks are almost impossible, designers should concentrate on methods to secure and protect the sensitive information from physical attacks. The paragraph below discusses possible solutions towards confirming the integrity of codes running in the sensor node and protecting highly sensitive data through Trusted Computing and TrustZone technology.

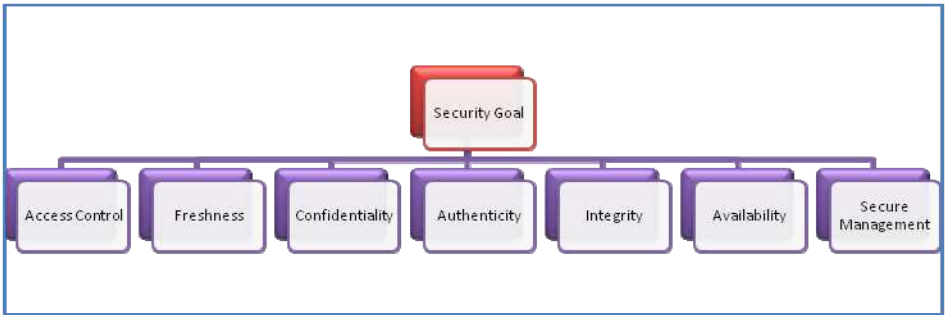


Fig. 1. Common Security Goals in Wireless Sensor Networks

4. 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 (Grawrock, 2009), trust can be defined as an entity that always behaves in the expected way for the intended function. Basic properties of a trusted computer or systems [referenced from?]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 the desired security level. Trusted Computing Platform Alliance (TCPA) was formed in late 90's and finally emerged as the Trusted Computing Group (TCG) in 2003 (Groups, 2008). TCG has basically worked to develop an inexpensive chip that helps users protect their sensitive information.

Muhammad Amin (Amin *et al.*, 2008b) in his paper discussed on the 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.

4.1 Trusted Platform Module

Trusted Computing Groups (TCG) 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 in software (Grobschadl *et al.*, 2008) where the configuration of TPM is vendor specific and is not specified by TCG. Fig. 2 briefly shows block diagram of TPM consisting of ten components to accelerate security processes.

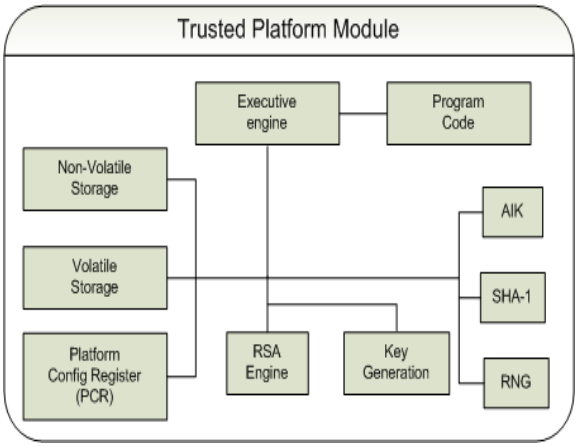


Fig. 2. Standard TPM Components

Unfortunately, the choice of RSA and SHA-1 algorithms has made the platform unsuitable for WSN applications. RSA with 2048 bits has been confirm to consume higher energy and therefore unsuitable for WSN applications and embedded system (Amin *et al.*, 2008a). Moreover, RSA when implemented in hardware demand large silicon area and therefore increase the size of the chip (Kocabas *et al.*, 2008b). 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 use in new design.

To conclude, TPM model may not be the best choice for secure or trusted platform implementation in embedded systems 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.

4.2 Trust Zone in ARM Microprocessor

The key feature of the 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 external security chip approach (Halfhill, 2003).

The 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 (Wilson *et al.*, 2007) 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.

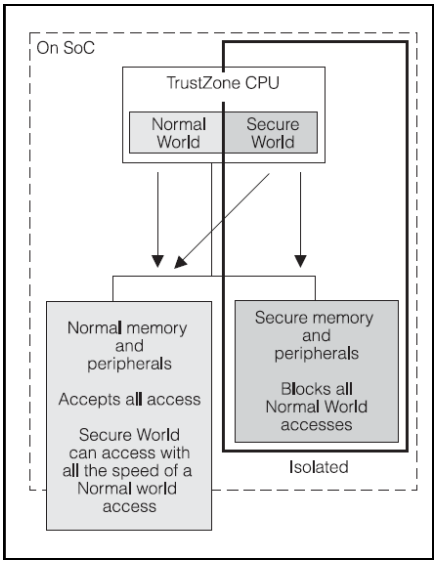


Fig. 3. One core support two operating worlds: secure world and normal world. Courtesy of: Wilson.P et.al (Wilson *et al.*, 2007)

The switching between secure and non-secure world in the ARM processor is established through the Secure Monitor Call (SMC) instruction and interrupts. In line with WSN constraints, the trust zone in the ARM processor eliminates the need for extra security chip. Moreover, security elements can be executed at full processor speed without cache-flushing overhead. It can also save the power as only one of the two virtual processors run at one time. Fig. 3 shows how trustzone mimics two processors.

5. Related Studies

G.Edward Suh et.al (Suh *et al*., 2007) 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, Physical Random Functions (PUFs) and off-chip memory functions.

Physical Random Function (PUFs) is a function that generates secret numbers 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. (Lie *et al*., 2000) 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 (Hu *et al*., 2009) which was mentioned earlier used external TPM chip on the sensor node. This TPM based public key platform facilitates message security services with 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. Besides the security algorithm used is not aligned with sensor node constraints.

Another work on hardware based security is done by (PANIANDI, 2006, Pin, 2009) 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(Winter, 2008) and Xu Yang-ling(Xu *et al.*, 2008), 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.

6. Proposed Work

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. Almost all available sensor node platforms (Healy *et al.*, 2008) 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 provide better shielding to private information on the platform.

6.1 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 from running and to ensure the confidentiality and integrity of the data during communications.

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

- Secure world – all the sensitive resources will be placed in the secure world memory locations. Trust zone Address space controller (TZASC) is used to configure regions as secure or non-secure. All non-secure process will be rejected to the 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 software images and devices on the platform.
- On-Soc RAM and ROM will ensure no highly sensitive data leaves the chip thus eliminating the possibility of physical attacks.
- * Identity based Encryption Algorithm for confidentiality and integrity of the data during communications. (Communications between sensor node and base station)

By using ARM trust zone, a small on-chip security system is presented in Fig. 4 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. Trust Zone 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.

*Not discussed in this paper.

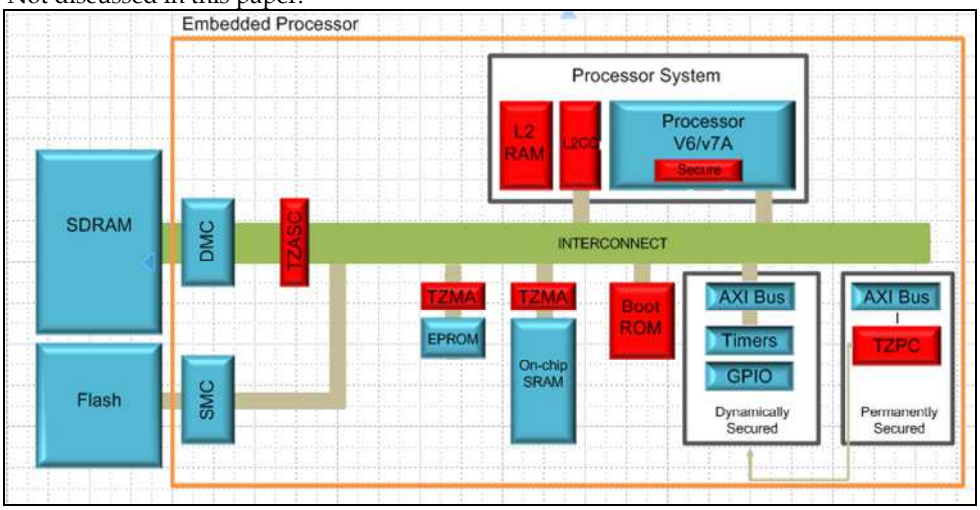


Fig. 4. Proposed security architecture for sensor node using ARM11 with Trust zone features.

Trust zone 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. 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 protocol such as secure boot will also be configured to complete the security design. Secure boot with the root of trust located in On-SoC ROM will provide a chain of trust for all the secure world software and hardware peripherals and some of the normal world software. With secure boot, the integrity of the OS image, software and peripherals on the platform can be verified to be truly unadulterated. Communications right after the secure boot process can be confirmed coming from a trusted sensor node.

Table 1 clearly depicts the advantage of the proposed security mechanism over previous work. Although the security level of the second technique is comparable with the proposed work, this proposed scheme offers 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. Moreover, AEGIS works is does not consider WSNs constraints. 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.

Previous Worked	Definition	Advantage	Drawback	Secure(S) Trusted (T)	Attacks Physical (PHY) Software (SW)	Consider WSN constraints?
External Hardware **TPM – RSA [3]** **TPM - IBE [18]** **AES - [5]** **RSA – [4][19]**	I 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 Physical attacks	T&S T&S S S	SW	**NO**
Embedded Hardware **AEGIS - AES[1]** **XOM- [2]**	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 on-chip components. Not flexible	T&S S	SW & PHY	**NO**
Embedded security H/W with Dual Virtual CPU (Trustzone (TZ)) **TZ+MTM [6]** **TZ+MAC [7]**	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. Flexible design-can secure up to off-chip components	For mobile appliances	T&S T&S	SW & PHY	**NO** **NO**
Proposed work **ARM11 with Trustzone**	**As above**	**As Above**	**For sensor node**	**T&S**	**SW & PHY**	**YES**

Table 1. Comparison Study on Trusted Implementation for Wireless Sensor Network

7. Conclusion

The security features discussed earlier are intended for highly secure applications dealing with crucial financial information, noncritical military communications, medical data, and critical

corporate information. Detail on security level can be found in (Groups, 2010). Two dominant features that differentiate this work from others are the placement of sensitive resources such as the crypto keys within the embedded system and the denial of extra or dedicated processor core for security purposes. This implementation ensures no sensitive resources leaves the chip and therefore blocks 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 (Vieira *et al.*, 2003).

8. References

Alves, T., D. Felton & ARM (2004): TrustZone: Integrated Hardware and Software Security. In *Technology in-Depth*: 18(Ed)^(Eds).

Amin, F., A. H. Jahangir & H. Rasifard (2008a): Analysis of Public-Key Cryptography for Wireless Sensor Networks Security. *World Academy of Science, Engineering and Technology***:** 529.

Amin, M., S. Khan, T. Ali & S. Gul (2008b): Trends and Directions in Trusted Computing: Models, Architectures and Technologies. In *International MultiConference of Engineers and Computer Scientists*(Ed)^(Eds). Hong Kong.

Becher, A., Z. Benenson & M. Dornself (2006): *Security in Pervasive Computing*: Springer Berlin/Heidelberg.

Ekanayake, V., I. Clinton Kelly & R. Manohar (2004): An ultra low-power processor for sensor networks. In *Proceedings of the 11th international conference on Architectural support for programming languages and operating systems*(Ed)^(Eds). Boston, MA, USA: ACM.

Gaubatz, G., J.-P. Kaps, E. Ozturk & B. Sunar (2005): State of the ART in Ultra Low Power Public key Cryptography for Wireless Sensor Network. In *3rd International Conference on Pervasive Computing and Communications Workshop*(Ed)^(Eds): IEEE Computer Society.

Grawrock, D. (2009): *Dynamics of a Trusted Platform*: Intel Press.

Grobschadl, J., T. Vejda & D. Page (2008): Reassassing the TCG Specifications for Trusted Computing in Mobile Embedded Systems. In *1st IEEE Workshop on hardware-Oriented Security and Trust HOST2008*: 84(Ed)^(Eds): IEEE.

Groups, E. T. (2010): Cryptography for embedded systems(Ed)^(Eds): EE Times Network.

Groups, T. C. (2008): Trusted Platform Module(TPM) Summary(Ed)^(Eds): Trusted Computing Groups.

Halfhill, T. R. (2003): ARM DONS ARMOR: Trustzone Security Extensions Strengthnen ARMv6 Architecture. In *MIcroprocessor*(Ed)^(Eds). Arizona: Reed Electronics Group.

Healy, M., T. Newe & E. Lewis (2008): Wireless Sensor Node hardware: A review. In *Sensors, 2008 IEEE*: 621(Ed)^(Eds).

Hu, W., P. Corke, W. C. Shih & L. Overs (2009): SecFleck: A public key technology platform for wireless sensor networks: 296(Ed)^(Eds). Cork, Ireland: Springer Verlag.

Huai, L., X. Zou, Z. liu & Y. Han (2009): An Energy Efficient AES-CCM Implementation for IEEE802.15.4 Wireless Sensor Networks. In *2009Internatioanal Conference on networks Security, Wireless Communications and Trusted Computing*: 394(Ed)^(Eds): IEEE Computer Society.

Huang, A. L. & W. T. Penzhorn (2005): Cryptographic Hash Functions and Low-Power Techniques for Embedded Hardware. In *Industrial Electronics, 2005. ISIE 2005. Proceedings of the IEEE International Symposium on*: 1789(Ed)^(Eds).

Kocabas, O., E. Sabas & J. Grobschadl (2008a): Enhancing an Embedded Processor Core with a Cryptographic Unit for Performance and Security In *4th International Conference on Reconfigurable Computing and FPGAs*: 409(Ed)^(Eds): IEEE.

Kocabas, O., E. Savas & J. Grossschadl (2008b): Enhancing an Embedded Processor Core with a Cryptographic Unit for Speed and Security. In *Reconfigurable Computing and FPGAs, 2008. ReConFig '08. International Conference on*: 409(Ed)^(Eds).

Lee, Y. K., k. Sakiyama, L. Batina & I. Verbauwhede (2008): Elliptic-Curve-Based Security processor for RFID(Ed)^(Eds): IEEE Computer Society.

Lie, D., C. Thekkath, M. Mitchell, P. Lincoln, D. Boneh, J. Mitchell & M. Horowitz (2000): Architectural support for copy and tamper resistant software. *SIGPLAN Not.* **35:** 168.

PANIANDI, A. (2006): *A Hardware Implementation of Rivest-Shamir-Adleman Co-Processor for Resource Constrainted Embedded Systems*. Master, Universii Teknologi Malaysia, Skudai.

Pathan, A. S. K. & H. Choong Seon (2008): Feasibility of PKC in resource-constrained wireless sensor networks. In *Computer and Information Technology, 2008. ICCIT 2008. 11th International Conference on*: 13(Ed)^(Eds).

Perrig, A., R. Szewczyk, J. D. Tygar, V. Wen & D. E. Culler (2002): SPINS: security protocols for sensor networks. *Wirel. Netw.* **8:** 521.

Pin, L. Y. (2009): *Verilog Design of a 256-bits AES Crypto Processor Core*. Master, Universiti Teknologi Malaysia, Skudai.

Suh, G. E., C. W. O'Donnell & S. Devadas (2005): AEGIS: A single-chip secure processor. *Information Security Technical Report* **10:** 63.

Suh, G. E., C. W. O'Donnell & S. Devadas (2007): Aegis: A Single-Chip Secure Processor. *IEEE Des. Test* **24:** 570.

Verma, N. (2006): *Practical Implementation and Performance Analysis On Security of Sensor Networks*. MSc Full Thesis, Rochester Institute of Technology, Rochester, New York.

Vieira, M. A. M., C. N. Coelho, Jr., D. C. da Silva, Jr. & J. M. da Mata (2003): Survey on wireless sensor network devices. In *Emerging Technologies and Factory Automation, 2003. Proceedings. ETFA '03. IEEE Conference*: 537(Ed)^(Eds).

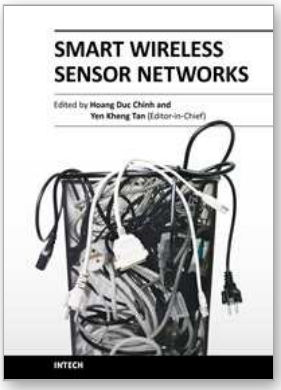
Wilson, P., A. Frey, T. Mihm, D. Kershaw & T. Alves (2007): Implementing Embedded Security on Dual-Virtual-CPU Systems. In *IEEE Design and Test of Computers*: 582(Ed)^(Eds): IEEE Computer Society.

Winter, J. (2008): Trusted computing building blocks for embedded linux-based ARM trustzone platforms. In *Proceedings of the 3rd ACM workshop on Scalable trusted computing*(Ed)^(Eds). Alexandria, Virginia, USA: ACM.

Xu, Y.-l., W. Pan & X.-g. Zhang (2008): Design and Implementation of Secure Embedded Systems Based on Trustzone. In *Embedded Software and Systems, 2008. ICESS '08. International Conference on*: 136(Ed)^(Eds).

Yong, W., G. Attebury & B. Ramamurthy (2006): A survey of security issues in wireless sensor networks. *Communications Surveys & Tutorials, IEEE* **8:** 2.

Znaidi, W., M. Minier & J.-P. Babau (2008): An Ontology for Attacks in Wireless Sensor Networks(Ed)^(Eds). Montbonnot Saint Ismier: National De Recherche En Informatique Et En Automatique.



Smart Wireless Sensor Networks

Edited by Yen Kheng Tan

ISBN 978-953-307-261-6
Hard cover, 418 pages
Publisher InTech
Published online 14, December, 2010
Published in print edition December, 2010

The recent development of communication and sensor technology results in the growth of a new attractive and challenging area â€“ wireless sensor networks (WSNs). A wireless sensor network which consists of a large number of sensor nodes is deployed in environmental fields to serve various applications. Facilitated with the ability of wireless communication and intelligent computation, these nodes become smart sensors which do not only perceive ambient physical parameters but also be able to process information, cooperate with each other and self-organize into the network. These new features assist the sensor nodes as well as the network to operate more efficiently in terms of both data acquisition and energy consumption. Special purposes of the applications require design and operation of WSNs different from conventional networks such as the internet. The network design must take into account of the objectives of specific applications. The nature of deployed environment must be considered. The limited of sensor nodesâ€™ resources such as memory, computational ability, communication bandwidth and energy source are the challenges in network design. A smart wireless sensor network must be able to deal with these constraints as well as to guarantee the connectivity, coverage, reliability and security of networkâ€™s operation for a maximized lifetime. This book discusses various aspects of designing such smart wireless sensor networks. Main topics includes: design methodologies, network protocols and algorithms, quality of service management, coverage optimization, time synchronization and security techniques for sensor networks.

How to reference

In order to correctly reference this scholarly work, feel free to copy and paste the following:

Yusnani Mohd Yussoff, Habibah Hashim and Husna Zainol Abidin (2010). Integrity Enhancement in Wireless Sensor Networks, Smart Wireless Sensor Networks, Yen Kheng Tan (Ed.), ISBN: 978-953-307-261-6, InTech, Available from: http://www.intechopen.com/books/smart-wireless-sensor-networks/integrity-enhancement-in-wireless-sensor-networks

INTECH
open science | open minds

InTech Europe
University Campus STeP Ri
Slavka Krautzeka 83/A
51000 Rijeka, Croatia
Phone: +385 (51) 770 447

InTech China
Unit 405, Office Block, Hotel Equatorial Shanghai
No.65, Yan An Road (West), Shanghai, 200040, China
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元
Phone: +86-21-62489820

Fax: +385 (51) 686 166
www.intechopen.com

Fax: +86-21-62489821