A SECURE OVR-THE-AIR PROGRAMMING SCHEME IN WIRELESS SENSOR NETWORKS

FARZAN DOROODGAR HEZAVEH

A project report submitted in partial fulfillment of the
requirements for the award of the degree of
Master Computer Science (Information Security)

Faculty of Computer Science and Information Systems
Universiti Teknologi Malaysia

JANUARY 2013

This thesis is dedicated to my mother Fereshteh Hendi, and sister Farzaneh Doroodgar who have supported me all the way since the beginning of my studies. I am also dedicating this thesis to three beloved people who have meant and continue to mean so much to me. Although they are no longer of this world, their memories continue to regulate my life. My Father Alireza Doroodgar, my maternal grandfather Gholamabbas Hendi and grandmother Roghayeh Hashtroodi that their love for me knew no bounds and they thought me more than I ever will learn. Also, this thesis is dedicated to my wife, Masoomeh Sadat Aleyasin who has been a great source of motivation and inspiration.

Finally I would like to dedicate my thesis to my Aunt Akram Hendi and my Uncle Alireza Hendi for their endless support during every step of my life and more brilliantly during my studies.

ACKNOWLEDGEMENT

First and foremost, I would like to express heartfelt gratitude to my supervisor **Dr. Mohammad Abdur Razzaque** for his constant support during my study at UTM. He inspired me greatly to work in this project. His willingness to motivate me contributed tremendously to our project. I have learned a lot from him and I am fortunate to have him as my mentor and supervisor. My sincere thanks also goes to Dr. Abu Bakar Kamalrulnizam and Dr. Majid Bakhtiari for their valuable feedbacks and corrections.

Besides, I would like to thank the authority of Universiti Teknologi Malaysia (UTM) for providing me with a good environment and facilities such as Computer laboratory to complete this project with software which I need during process.

ABSTRACT

Over-The-Air dissemination of code updates in Wireless Sensor Networks (WSNs) have been researchers' point of interest in past a few years and more importantly security challenges toward remote propagation of code update have taken the majority of efforts in this context. Many security models have been proposed to establish a balance between the energy consumption and security strengthen with having their concentration on constraint nature of WSN nodes. For authentication purposes most of them have used Merkle-Hash-Tree to avoid using multiple public cryptography operations. These models mostly have assumed an environment in which security has to be in a standard level and therefore they have not investigated the tree structure for mission-critical situations in which security has to be in maximum possible extent (e.g. military zones). Two major problems have been identified in Merkle Tree structure which is used in Seluge scheme, including: 1) an exponential growth in number of overhead packets when block size of hash algorithm used in design is increased. 2) Limitation of using hash algorithms with larger block size of 11 bytes when payload size is set to 72 bytes. Then several existing security models are investigated for possible vulnerabilities and a set of countermeasures correspondingly named Security Model Requirements (SMR) is provided. After concentrating on Seluge's design, a new secure Over-The-Air Programming (OTAP) scheme named Seluge++ is proposed that complies with SMR and replaces the use of inefficient Merkle Tree with a novel method.

ABSTRAK

Kemaskini kod bagi penyebaran melalui udara di dalam Rangkaian Sensor Tanpa Wayar (WSN) telah menjadi perhatian para pengkaji di dalam bidang ini sejak beberapa tahun yang lalu. Perkara yang paling penting adalah cabaran-cabaran terhadap keselamatan ke atas propagasi secara kawalan bagi kemaskini kod yang telah mengambil sebahagian daripada usaha tersebut. Terdapat banyak model-model keselamatan telah dicadangkan bagi mewujudkan keseimbangan antara penggunaan tenaga dan juga kekuatan keselamatan dengan memfokuskan kepada kekangan nod-nod WSN. Bagi tujuan pengesahan pula, kebanyakan mereka telah menggunakan Merkle-Hash-Tree bagi mengelakkan penggunanaan operasi kiptografi am yang pelbagai. Kebanyakan model-model tersebut beranggapan bahawa keselamatan perlu berada pada tahap piawai yang sepatutnya. Oleh itu, ia tidak mengkaji struktur pokok bagi situasi kritikal di mana tahap keselamatan perlu berada pada tahap yang paling maksimum contohnya di kawasan tentera. Dua masalah utama yang telah dikenalpasti oleh Merkle Tree yang digunapakai di dalam skim Seluge adalah: 1) penambahan paket secara mendadak apabila saiz blok algoritma hash yang digunakan meningkat. 2) kekangan dalam penggunaaan algoritma hash dengan saiz blok yang lebih besar dari 11 bait apabila saiz muatan disetkan kepada 72 bait. Beberapa lagi model keselamatan yang lain telah mengkaji beberapa kelemahan yang berpotensi dan juga langkah-langkah pencegahan turut disediakan dan diberi nama sebagai Keperluan Model Keselamatan (SMR). Selepas memfokuskan kepada rekabentuk Seluge, satu skim Pengaturacaraan Over-The-Air (OTAP) yang lebih selamat dan dikenali sebagai Seluge++ telah dicadangkan di mana ia telah mematuhi piawaian SMR dan juga dapat menggantikan penggunaan Merkle Tree yang tidak berapa berkesan dengan kaedah novel.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

1.1. Background of Study

With the growing impact of wireless communications and the facilities they provide in many different fields of our life, they have been employed to integrate into many devices where wired communications are not practical. One of the emerging areas of widely using wireless communications is Wireless Sensor Networks (WSNs) which are distributed among different areas like military zones, medical science, environmental science, geography and etc.

Generally, Wireless Sensor Networks do not have any specific topology and are consisting of hundreds or thousands of nodes wirelessly connected to each other capable of sensing environment and also processing data. A node also known as mote ideally has a programmable microcontroller as its processing unit, at least one sensor to sense the environmental situations to convert it into digital format (e.g. sensing temperature, humidity) and also a limited power source which in most cases is a battery. To be more precise a mote can always be a node but a node must have at least one sensor to be called a mote. For example some nodes are just used within a WSN to route the packets (clustering) and these nodes do not take part in data gathering of motes.

Although nodes are very limited in resources like their power source and processing capabilities, WSNs are mostly required to be long live and retrieve and process data in real-time, these networks are commonly used in scenarios where using ad-hoc or wired is not economical, number of nodes is spontaneously large, and also there are physical access difficulties to the target environment so it's almost impossible to have physical access to the deployed sensor nodes after distributing them.

In software engineering perspective, it is very likely for nodes to encounter some bugs in their installed firmware or they might need some new functionality to be placed into their firmware. Considering above mentioned facts, the following questions raise:

- How to update installed firmware on nodes, where there are hundreds or thousands of nodes established in a network with physical access limitations?
- How should the responsible protocol or model work to provide an efficient mechanism to update nodes, regarding all the limitations of WSNs mentioned in previous statements?
- What specific criteria have to be included in update scheme of WSNs?

There are two classifications of updates (Sreenan & Brown, 2006): *Static* which needs the target node to be restarted, in this mode update itself might be incremental or monolithic. *Dynamic* update is another applicable update classification in WSNs in which the node does not necessarily need to be restarted and just an interruption may occur in execution of target software. For dynamic reprogramming there exists two sub-categories (Galos, Mieyeville, Navarro, & O'Connor, 2011) including partial and full that regardless of how they work, Their main important difference in our context is the time needed for them to take effect. In some cases like military zones motes are placed close to the adversary and therefore it is much easier for attackers to have physical access to nodes more than network owners, so no matter static or dynamic update is chosen, it must be possible to update nodes over-the-air with the smallest possible time. This functionality not

only removes the need for physical access to nodes which is almost impossible in many cases but also will make it very easy to update the whole network with numerous nodes. Remote update itself is suffering from a few problems in its nature (Sreenan & Brown, 2006):

- Possible interference with the default communication structure which is used to transmit data;
- The chance of a partial or full failure of network if any fault occur within an upgrade
- Updates costs might reduce lifetime of the networks.

To reduce impact of these problems there have been many efforts in recent years, for example to reduce the size of an update image file (Galos et al., 2011) have proposed MinTax which is a high-level language compiled on the node. They have proposed usage of delta files instead of complete binary image file. Another research done in (Schroder-Preikschat et al., 2007) takes care of heterogeneities of WSNs and by proposing a profile based software management scheme reduces the size of transmitted data while reprogramming.

In 2006, a new model has been proposed by (Sreenan & Brown, 2006) to cover all the requirements of update software, this model is not considered a full-design and only provides fundamental basics needed to implement software update. A set of criteria with their necessity is provided which are required for every model to have them in design. Following is a list of these criterions with a quick review of their importance bolded by (Sreenan & Brown, 2006):

- **Functionality** – Is consist of three main steps needed before an update starts executing on a node: *Generation* is action of creating an update on the host administrative system, *Propagation* is the action of transferring generated update files into network so that target nodes will receive, download and keep it for further step; and *Activation* is the act of replacing currently running code inside memory with the new downloaded image files and

marking it as available for execution. The benefit of existence of these phases is that feedback methods can be used to improve further updates on particular networks.

- **Performance** – The reprogramming scheme running on WSNs must provide efficient and robust management of available resources so that it will not cause great reduction in lifetime of networks.
- **Reliability** – Regarding basic properties of WSNs especially the likelihood of large-scale deployment and also difficulties for physical access to the nodes, it's very essential for the update scheme to be reliable. Update management software like any other software might encounter some run-time errors which must take care of them independently. Proposed scheme must be able to rollback to any known good image file in case of any unhandled exception. Another important aspect of updating nodes is that by nature they don't have any specific topology so it's very likely for some nodes to get disconnected from the network and after a while get back into the network. This incident raises another problem in which proposed scheme must be reliable in these cases so that if a number of nodes are not available at the time update is pushed into the system, later after they get online it must be guaranteed that outdated nodes will instantly be updated without old code getting executed unless it's allowed to do so.
- **Usability** – Apart from possibility of update for OS and firmware for a node, software update mechanism must also provide an update structure for itself. Versioning will be a key factor in the criterion.
- **Portability** – Proposed mechanisms must be compatible with different hardware, middleware, MAC, and infrastructures of WSNs.
- **Security** – It is our focus in this report, WSNs are used in very important fields where security is playing a key role in their usage (e.g. military zones). In these area there are sensitive information transferred within the network that an intruder can easily intercept them and therefore gain unauthorized access to system. *Authentication*, *Privacy preservation*, *Data Integrity*, and *DoS attack countermeasures* are generic concepts needed to be considered as a set of security-critical requirements to preserve known security concepts: Confidentiality, Integrity and Availability (CIA).

Regarding specific nature of WSNs with its unique structure and limitations, security concepts in WSNs are facing different problems apart from related ones to powerful networks like Internet. In this report the focus will only be on over-the-air update security problems which include the following issues:

- What happens if an attacker replaces trusted software installed on nodes with a malicious code?
- What if an attacker intercept binary image file while transmission and extract sensitive data out of it? (E.g. passwords, shared keys, etc.)
- Is it possible for attackers to modify binary image file with their malicious content while it's transmitting over the air?
- If an attacker tries to send a lot of update request to nodes, does it affect normal behavior of network?

Current existing reprogramming schemes for WSNs that are designed so that an update will quickly pushed into nodes, although this is useful in term of resource management because it will reduce transmission time, but it makes the network very susceptible to the attacks where a security hole is found specially in software update mechanism; an attacker compromising a single node can easily compromise the whole network in short period of time.

A malicious code pushed into the network by an attacker can take the whole network down or more wisely can be used to spread invalid data to corrupt the results created by network. Consequently, the need of having a secure remote update scheme in WSN is highly important.

Wireless Sensor Networks are used in many areas whereas security is playing a key role and is an important issue (Fan & Gong, 2012), in places like military zones (Li, Batten, & Doss, 2009), industry automation and healthcare systems (Chien, Chan, Vu Chien, Nguyen Chan, & Nguyen Huu, 2011). Furthermore, wireless medium used as communication channel in WSN is considered to be an unsecure and untrusted way of communication in which an attacker can easily

eavesdrop, inject, delay, modify or remove any packet transmitted through network (He, Member, Chen, Chan, & Bu, 2012). In particular, Use of wireless medium and inherent collaborative nature of the network protocols make such network vulnerable to various forms of attacks (Maheshwari, Gao, & Das, 2007).

In result, although many OTAP protocols have been proposed (e.g. Deluge (Hui & Culler, 2004), MNP (Kulkarni, 2005), MOAP (T Stathopoulos, Heidemann, & Estrin, 2003), Aqueduct (Schroder-Preikschat et al., 2007), etc.) but most assume non-malicious nodes. Considering that this assumption is inadequate for many types of applications, four security aspects (Confidentiality, Integrity, Authentication and Availability) are required in OTAP protocols as authors of (De la Parra & Garcia-Macias, 2009) suggest. Therefore several vulnerabilities exist when an over-the-air programming (OTAP) protocol wants to propagate an update code remotely. An eavesdropper can threaten *Confidentiality* by gaining information by sniffing image data being disseminated to nodes(Bui, Ugus, Dissegna, Rossi, & Zorzi, 2010). *Availability* of network is threatened because the adversary may inject bogus packets during the code propagation to force sensor nodes to propagate the corrupted image potentially over multiple hops to deplete their limited power (Ugus, Westhoff, & Bohli, 2009). An adversary can inject a malicious update into the system. Given the epidemic nature of network reprogramming protocols, an adversary can gain complete control over the entire sensor network by compromising just a single node (Lanigan, Gandhi, & Narasimhan, 2006), therefore *Authenticity* of network is threatened. *Integrity* can also be threatened if an adversary modifies update packets before they arrive at the receiver end. Additionally, Replay attacks, Battery-Drain attacks and Wormhole attacks are still possible on OTAP protocols and their existing proposed security models (Aschenbruck, Bauer, Bieling, Bothe, & Schwamborn, 2012; Hu, Tan, Corke, Shih, & Jha, 2010; Hyun, Ning, Liu, & Du, 2008; Lanigan et al., 2006; Li et al., 2009; Maheshwari et al., 2007; Perrig & Johnson, 2006).

Given the situation, Security is a feature that will need to be supported in any serious reprogramming system (De la Parra & Garcia-Macias, 2009). The security attributes for security sensitive applications in ad hoc networks (authentication, integrity, confidentiality and availability) are well defined (Zhou & Haas, 1999), and

apply to the particular case of WSN. The challenge to achieve this goal is the severe resource constraints of WSNs, namely the limited memory, energy, bandwidth, and processing. Therefore, the overhead caused on a secure OTAP scheme has to be minimum compared to the security strengthens achieved.

1.2. Statement of Problem

For authentication and also integrity purposes in OTAP protocols, many security models have used hash dependency in their design but with different patterns of constructing hash values (Dutta, Hui, Chu, & Culler, 2006; Hyun et al., 2008; Lanigan et al., 2006; Law, Zhang, Jin, Palaniswami, & Havinga, 2011). Many of them use Mekle Hash Tree (Deng, Han, Mishra, Dengcoloradoedu, & Hancoloradoedu, 2006) in a common scenario in which they have assumed a perfect tree and did not investigate the tree structure itself which may result in an inefficient tree structure (Kondratieva & Seo, 2007). Furthermore, Seluge (Hyun et al., 2008) is the best security model compared to others due to the fact that its authors claim that they have implemented all the security concepts in their design (For more information refer to comparison of all models at the end of Chapter 2). However it is also using Merkle Hash Tree that consequently will result in having the inefficiencies described above by default in its design.

1.3. Purpose of the Study

As explained in section 1.2, update mechanism for WSNs is a critical part of these networks with some specific criteria that must be included in any update scheme deployed for WSNs. The most important criterion is the security attribute which makes these sensors trusted so that their usages in mission-critical situations like human health related issues or military zones would be undoubtedly possible. Recent studies in this field as will be discussed in Chapter 2, show that current

reprogramming schemes are not fully resistant to security vulnerabilities like DoS, Man-In-The-Middle, Replay, Battery-Drain and Wormhole attacks (Deng et al., 2006; Lanigan et al., 2006; Munivel & Ajit, 2010). The proposed schemes either partially support security countermeasures or they are left fully unprotected(Hui & Culler, 2004). Some of them also are encountering high overhead of packets needed to be sent over air which will result in more power required to send these packets.

As result, this research has been started to cover the current security problems in reprogramming schemes so that all the security concepts (CIA) could be preserved with minimum overhead, as well as considering a mitigation mechanism to DoS attacks, Replay attacks, Wormhole attacks and Battery-Drain attacks against reprogramming mechanism.

1.4. Objectives of the Study

- To survey secure Over-The-Air Programming (OTAP) schemes in WSNs
- To improve security and decrease overhead in OTAP schemes in WSNs
- To test and validate proposed scheme in simulation tools (e.g. TOSSIM).

1.5. Research Questions

- What are vulnerabilities in existing secure OTAP protocols in WSNs?
- How to make sure current secure OTAP schemes fully support authentication and are protected against attacks which might cause threats to Confidentiality, Integrity and Availability?

1.6. Significance of Study

Existing security vulnerabilities in WSNs caused authorized authorities to hesitate in using WSNs in mission-critical situations and their results are not very trusted where human health might be under the risk. Although current security countermeasures applied in reprogramming schemes are partially protecting these networks but still attackers can easily invalidate the results generated by a network and when these results for example are going to be used by a military analyzer it can cause heavy damages. Therefore results of this study will help governments or security agencies to decide better on where and how to use WSNs especially when the protection against external attacks is highly important.

1.7. Scope of the Study

The general idea of this project is to manipulate a secure re-programming method for remotely updating WSN nodes over the air so that security concepts like confidentiality, availability and integrity would be preserved. This requires to fully understand chosen update model and also to implement required security countermeasures on top layers of communication and infrastructure of this method. Update procedures not only must be secure but also must be reliable enough to have a broadcasting update scheme which will take care of all the nodes whether by the moment they are in range or not.

1.8. Contribution of the Study

The existing vulnerabilities in OTAP protocols needed an efficient security model. But many security models that have been proposed so far are lacking some key concepts in their design. Seluge, one of these models, have an acceptable level of security countermeasure in its design that will make it suitable security model for

most of the scenarios. But when it comes to a mission critical situation like military zones it will fail because it is using Merkle Hash Tree that has inefficiencies as described in Section 1.2. In this research, two major inefficiencies in structure of Merkle Hash Tree are identified, which are:

1. Security limitation of hash algorithms with block size larger than a specific number (For example, Seluge can not afford using SHA-1 in its design).
2. High overhead for number of packets that is required to be sent as hashing block size is increased to provide more security strengthen.

Finally, Seluge++ is proposed that not only will leave security strengthen of Seluge intact but also it solves its inefficiencies and in addition it provides countermeasures to Replay, Battery-Drain and Wormhole attacks. Therefore Seluge++ will be the best alternative to be used in situations where WSN nodes are going to be deployed in a hostile environment like battlefields.

REFERENCES

Adekunle, a. a., & Woodhead, S. R. (2009). On Efficient Data Integrity and Data Origin Authentication for Wireless Sensor Networks Utilising Block Cipher Design Techniques. *2009 Third International Conference on Next Generation Mobile Applications, Services and Technologies*, 419–424. Ieee. Retrieved January 7, 2013, from http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=5337392

Aiash, M., Mapp, G., Phan, R. C.-W., Lasebae, A., & Loo, J. (2012). A Formally Verified Device Authentication Protocol Using Casper/FDR. *2012 IEEE 11th International Conference on Trust, Security and Privacy in Computing and Communications*, 1293–1298. Ieee. Retrieved from http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6296128

Aschenbruck, N., Bauer, J., Bieling, J., Bothe, A., & Schwamborn, M. (2012). Selective and Secure Over-The-Air Programming for Wireless Sensor Networks. *2012 21st International Conference on Computer Communications and Networks (ICCCN)*, 1–6. Ieee. Retrieved from http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6289278

Brown, S., & Sreenan, C. J. (2006). Updating software in wireless sensor networks: A survey. *Dept. of Computer Science, National Univ. of Ireland, Maynooth, Tech. Rep*. Citeseer. Retrieved June 29, 2012, from http://www.mendeley.com/research/updating-software-in-wireless-sensor-networks-a-survey/

Bui, N., Ugus, O., Dissegna, M., Rossi, M., & Zorzi, M. (2010). An integrated system for secure code distribution in Wireless Sensor Networks. *2010 8th IEEE International Conference on Pervasive Computing and Communications Workshops (PERCOM Workshops)*, 575–581. Ieee. Retrieved from http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=5470503

Cao, Q., Abdelzaher, T., Stankovic, J., & He, T. (2008). The LiteOS Operating System: Towards Unix-Like Abstractions for Wireless Sensor Networks. *2008 International Conference on Information Processing in Sensor Networks (ipsn 2008)*, 233–244. Ieee. Retrieved April 16, 2012, from http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=4505477

Chien, T. V., Chan, H. N., Vu Chien, T., Nguyen Chan, H., & Nguyen Huu, T. (2011). A comparative study on operating system for Wireless Sensor Networks. *Advanced Computer Science and Information System (ICACSIS), 2011 International Conference on* (pp. 978–979). Retrieved June 29, 2012, from http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=6140770

Christie, R. (1984). *Electricity, industry, and class in South Africa*. SUNY Press.

Chung-Wei Phan, R., Average, F. F., & Lee, S. (2002). Cryptanalysis of full Skipjack block cipher. *Electronics Letters*, *38*(2), 69–71. IET.

Crossbow. (n.d.). Mica2 Datasheet. Retrieved from http://bullseye.xbow.com:81/Products/Product_pdf_files/Wireless_pdf/MICA2_Datasheet.pdf

Das, M. (2008). Dynamic program update in wireless sensor networks using orthogonality principle. *Communications Letters, IEEE*, *12*(6), 471–473. Retrieved June 29, 2012, from http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=4542786

Deng, J., Han, R., Mishra, S., Dengcoloradoedu, J., & Hancoloradoedu, R. (2006). Secure code distribution in dynamically programmable wireless sensor networks. *Proceedings of the 5th international conference on Information processing in sensor networks* (pp. 292–300).

Diffie, W., & Hellman, M. (1976). New directions in cryptography. *Information Theory, IEEE Transactions on*, *22*(6), 644–654. IEEE.

Dong, W., Chen, C., Liu, X., & Bu, J. (2010). Providing OS support for wireless sensor networks: Challenges and approaches. *Communications Surveys &*, *12*(4), 519–530. IEEE. Retrieved June 29, 2012, from http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=5462978

Dunkels, A., Gronvall, B., & Voigt, T. (2004). Contiki - a Lightweight and Flexible Operating System for Tiny Networked Sensors. *Local Computer Networks, 2004. 29th Annual IEEE International Conference on* (pp. 455–462).

Dutta, P. K., Hui, J. W., Chu, D. C., & Culler, D. E. (2006). Securing the Deluge network programming system. *2006 5th International Conference on*

Information Processing in Sensor Networks, 326–333. ACM Press. Retrieved from http://portal.acm.org/citation.cfm?doid=1127777.1127826

Failures-Divergence Refinements – FDR2 user manual. (n.d.). . Retrieved January 13, 2013, from http://www.fsel.com/

Fan, X., & Gong, G. (2012). Accelerating signature-based broadcast authentication for wireless sensor networks. *Ad Hoc Networks*, *10*(4), 723–736. Elsevier B.V. Retrieved December 18, 2012, from http://linkinghub.elsevier.com /retrieve/pii/S157087051100148X

Galos, M., Mieyeville, F., Navarro, D., & O'Connor, I. (2011). Reprogramming hardware-software heterogeneous Wireless Sensor Networks. *Wireless Personal Multimedia Communications (WPMC), 2011 14th International Symposium on* (pp. 1–5). Retrieved June 29, 2012, from http:// ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=6081547

Gay, D., Levis, P., Von Behren, R., Welsh, M., Brewer, E., & Culler, D. (2003). The nesC language: A holistic approach to networked embedded systems. *Acm Sigplan Notices* (Vol. 38, pp. 1–11).

Han, C. C., Kumar, R., Shea, R., Kohler, E., & Srivastava, M. (2005). A Dynamic Operating System for Sensor Nodes. *Proceedings of the 3rd international conference on Mobile systems, applications, and services*, 163–176.

Handschuh, H., Knudsen, L. R., & Robshaw, M. J. (2001). Analysis of SHA-1 in Encryption Mode, 70–83.

He, D., Member, S., Chen, C., Chan, S., & Bu, J. (2012). SDRP : A Secure and Distributed Reprogramming Protocol for Wireless Sensor Networks, *59*(11), 4155–4163.

Housley, R., Yee, P., & Nace, W. (2000). *Encryption using KEA and SKIPJACK*.

Hu, W., Tan, H., Corke, P., Shih, W. C., & Jha, S. (2010). Toward trusted wireless sensor networks. *ACM Transactions on Sensor Networks (TOSN)*, *7*(1), 5. ACM. Retrieved June 29, 2012, from http://portal.acm.org /citation.cfm?doid=1806895.1806900

Hui, J. W., & Culler, D. (2004). The Dynamic Behavior of a Data Dissemination Protocol for Network Programming at Scale Categories and Subject Descriptors. *Proceedings of the 2nd international conference on Embedded networked sensor systems*, 81–94.

Hyun, S., Ning, P., Liu, A., & Du, W. (2008). Seluge: Secure and DoS-Resistant Code Dissemination in Wireless Sensor Networks. *2008 International Conference on Information Processing in Sensor Networks (ipsn 2008)*, 445–456. Ieee. Retrieved December 18, 2012, from http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=4505494

Karlof, C., Sastry, N., & Wagner, D. (2004). TinySec: a link layer security architecture for wireless sensor networks. *Proceedings of the 2nd international conference on Embedded networked sensor systems* (pp. 162–175).

Kondratieva, V., & Seo, S. (2007). Optimized Hash Tree for Authentication in Sensor Networks. *IEEE Communications Letters*, *11*(2), 149–151. Retrieved from http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=4115145

Kong, J. H., Ang, L.-M., Seng, K. P., & Ong, F. T. (2011). Low-complexity Two Instruction Set Computer architecture for sensor network using Skipjack encryption. *The International Conference on Information Networking 2011 (ICOIN2011)*, 472–477. Ieee. Retrieved from http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=5723161

Krawczyk, H., Canetti, R., & Bellare, M. (1997). HMAC: Keyed-Hashing for Message Authentication. Retrieved January 7, 2013, from http://tools.ietf.org/html/rfc2104

Kulkarni, S. S. (2005). MNP: Multihop Network Reprogramming Service for Sensor Networks. *25th IEEE International Conference on Distributed Computing Systems (ICDCS'05)*, 7–16. Ieee. Retrieved from http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=1437066

De la Parra, C. F. C., & Garcia-Macias, J. A. (2009). A protocol for secure and energy-aware reprogramming in WSN. *Proceedings of the 2009 International Conference on Wireless Communications and Mobile Computing Connecting the World Wirelessly - IWCMC '09*, 292. New York, New York, USA: ACM Press. Retrieved from http://portal.acm.org/citation.cfm?doid=1582379.1582443

Lanigan, P. E., Gandhi, R., & Narasimhan, P. (2006). Sluice: Secure Dissemination of Code Updates in Sensor Networks. *26th IEEE International Conference on Distributed Computing Systems (ICDCS'06)*, 53–53. Ieee. Retrieved from http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=1648840

Law, Y., Zhang, Y., Jin, J., Palaniswami, M., & Havinga, P. (2011). Secure Rateless Deluge: Pollution-Resistant Reprogramming and Data Dissemination for Wireless Sensor Networks. *EURASIP Journal on Wireless Communications and Networking*, *2011*(1), 685219. Retrieved November 15, 2012, from http://jwcn.eurasipjournals.com/content/2011/1/685219

Li, B., Batten, L. M., & Doss, R. (2009). Lightweight Authentication for Recovery in Wireless Sensor Networks. *2009 Fifth International Conference on Mobile Ad-hoc and Sensor Networks*, 465–471. Ieee. Retrieved December 18, 2012, from http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=5401475

Liu, A., & Ning, P. (2008). TinyECC: A configurable library for elliptic curve cryptography in wireless sensor networks. *Information Processing in Sensor Networks, 2008. IPSN'08. International Conference on* (pp. 245–256). Retrieved June 29, 2012, from http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=4505478

Liu, W., Luo, R., & Yang, H. (2009). Cryptography Overhead Evaluation and Analysis for Wireless Sensor Networks. *2009 WRI International Conference on Communications and Mobile Computing*, 496–501. Ieee. Retrieved January 7, 2013, from http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=4797303

Lowe, G. (1997). Casper: a compiler for the analysis of security protocols. *Proceedings 10th Computer Security Foundations Workshop*, 18–30. IEEE Comput. Soc. Press. Retrieved from http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=596779

Maheshwari, R., Gao, J., & Das, S. R. (2007). Detecting Wormhole Attacks in Wireless Networks Using Connectivity Information. *IEEE INFOCOM 2007 - 26th IEEE International Conference on Computer Communications*, 107–115. Ieee. Retrieved from http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=4215603

Michail, H. E., Kakarountas, a. P., Milidonis, a., & Goutis, C. E. (2004). Efficient implementation of the keyed-hash message authentication code (HMAC) using the SHA-1 hash function. *Proceedings of the 2004 11th IEEE International Conference on Electronics, Circuits and Systems, 2004. ICECS 2004.*, 567–570. Ieee. Retrieved from http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=1399744

Michail, H., Kakarountas, a. P., Koufopavlou, O., & Goutis, C. E. (2005). A Low-Power and High-Throughput Implementation of the SHA-1 Hash Function. *2005 IEEE International Symposium on Circuits and Systems*, 4086–4089. Ieee. Retrieved from http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=1465529

Munivel, E., & Ajit, G. M. (2010). Efficient Public Key Infrastructure Implementation in Wireless Sensor Networks. *2010 International Conference on Wireless Communication and Sensor Computing (ICWCSC)*, 1–6. Ieee. Retrieved from http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=5415904

Ning, P., Liu, A., & Du, W. (2008). Mitigating DoS attacks against broadcast authentication in wireless sensor networks. *ACM Transactions on Sensor Networks*, *4*(1), 1–35. Retrieved from http://portal.acm.org/citation.cfm?doid=1325651.1325652

Park, K., Lee, J., Kwon, T., & Song, J. (2007). Supplementary Hash in Wireless Sensor Networks, 653–662.

Perrig, a., & Johnson, D. B. (2006). Wormhole attacks in wireless networks. *IEEE Journal on Selected Areas in Communications*, *24*(2), 370–380. Retrieved from http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=1589115

Phillips, L. A. (2005). *Aqueduct: Robust and efficient code propagation in heterogeneous wireless sensor networks*. University of Colorado. Retrieved June 29, 2012, from http://www-users.cs.umn.edu/~phillips/Aqueduct.pdf

Pura, M.-L., & Patriciu, V.-V. (2010). Security analysis of Robust User Authentication Protocol. *2010 8th International Conference on Communications*, 457–460. Ieee. Retrieved from http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=5509078

Ramadass, S., Budiarto, R., & Ho, C. L. (2007). Unreliable Network Re-Authentication Protocol Based On Hybrid Key Using CSP Approach. *IJCSN International Journal Of Computer Science And Network Security*, *1*(11). Dr. Sang H. Lee.

Schneider, S. A., Goldsmith, M. H., Lowe, G., & Roscoe, A. W. (2010). The Modelling and Analysis of Security Protocols : the CSP Approach, (December).

Schroder-Preikschat, W., Kapitza, R., Kleinoder, J., Felser, M., Karmeier, K., Labella, T. H., & Dressler, F. (2007). Robust and efficient software

management in sensor networks. *Communication Systems Software and Middleware, 2007. COMSWARE 2007. 2nd International Conference on* (pp. 1–6). Retrieved June 29, 2012, from http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=4268114

Shnayder, V, Hempstead, M., Chen, B., & Welsh, M. (2004). Powertossim: Efficient power simulation for tinyos applications.

Shnayder, Victor, Hempstead, M., Chen, B., Allen, G. W., & Welsh, M. (2004). Simulating the power consumption of large-scale sensor network applications. *Proceedings of the 2nd international conference on Embedded networked sensor systems - SenSys '04*, 188. New York, New York, USA: ACM Press. Retrieved from http://portal.acm.org/citation.cfm?doid=1031495.1031518

Sreenan, C. J., & Brown, S. (2006). A new model for updating software in wireless sensor networks. *Network, IEEE*, *20*(6), 42–47. IEEE.

Stathopoulos, T, Heidemann, J., & Estrin, D. (2003). *A remote code update mechanism for wireless sensor networks*.

Stathopoulos, Thanos, Heidemann, J., & Estrin, D. (2003). A Remote Code Update Mechanism for Wireless Sensor Networks. CALIFORNIA UNIV LOS ANGELES CENTER FOR EMBEDDED NETWORKED SENSING. Retrieved June 29, 2012, from http://oai.dtic.mil/oai/oai?verb=getRecord&metadataPrefix=html&identifier=ADA482887

Su, M., Yang, X., Wei, L., & Yang, H. (2010). Key Management Scheme in WSN Based on Property of Circle. *2010 International Conference on Computational Intelligence and Software Engineering*, 1–4. Ieee. Retrieved from http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=5676908

TinyOS. (n.d.). . Retrieved from http://www.tinyos.net

Ugus, O., Westhoff, D., & Bohli, J. M. (2009). A ROM-friendly secure code update mechanism for WSNs using a stateful-verifier τ-time signature scheme. *Proceedings of the second ACM conference on Wireless network security* (pp. 29–40). Retrieved June 29, 2012, from http://dl.acm.org/citation.cfm?id=1514279

Wander, A. S., Gura, N., Eberle, H., Gupta, V., & Shantz, S. C. (2005). Energy analysis of public-key cryptography for wireless sensor networks. *Pervasive Computing and Communications, 2005. PerCom 2005. Third IEEE International Conference on* (pp. 324–328).

Zeng, P., Cao, Z., Choo, K. K. R., & Wang, S. (2009). Security weakness in a dynamic program update protocol for wireless sensor networks. *Communications Letters, IEEE*, *13*(6), 426–428. IEEE. Retrieved June 29, 2012, from http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=5090425

Zhou, L., & Haas, Z. J. (1999). Securing ad hoc networks. *Network, IEEE*, *13*(6), 24–30. IEEE.