

LB /Don /106 /2016

IT 01/133

**Design and Development  
of  
an Efficient and Secure Lightweight Protocol  
for  
Wireless Sensor Networks**

LIBRARY  
UNIVERSITY OF MORATUWA, SRI LANKA  
MORATUWA

K.Kesavan  
(139168 U)

Dissertation submitted to the Faculty of Information Technology,  
University of Moratuwa, Sri Lanka, for the partial fulfilment of the requirements  
of the Degree of Master of Science in Information Technology.

University of Moratuwa  
  
TH3168

March 2016

004<sup>16</sup>  
004 (043)

TH 3168

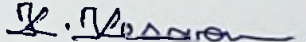
+  
1 DVD ROM  
(TH 3160 - TH 3180)

TH 3168

# Declaration

We declare that this thesis is our own work and has not been submitted in any form for another degree or diploma at any university or other institution of tertiary education. Information derived from the published or unpublished work of others has been acknowledged in the text and a list of references is given.

K. Kesavan.  
Name of Student (s)

  
Signature of Student (s)


Date : 28/04/2016

Supervised by

M. F. M. Firdhous .

Name of Supervisor(s)

***UOM Verified Signature***

  
Signature of Supervisor(s)

Date : 28/04/2016

*Dedicated to*  
*my teacher Mr. S. Thirunavukkarasu*



## **Acknowledgement**

It should be mentioned with gratitude that many individuals influenced to accomplish this project in many ways.

First of all, I would like to respectfully express my sincere gratitude to my supervisor, Mr. M.F.M.Firdhous (Senior Lecturer), for his guidance, insightful suggestions, invaluable comments, and constant encouragement and optimism that not only helped me overcome frustrations and difficulties throughout my project, but also will influence my future career in pursuit of excellence. I was really fortunate to complete my project under his supervision.

I am also indebted to the evaluators of the interim report of this project, Dr. L. Ranathunga, Dr. C.R.J. Amalraj and Mr. B.H. Sudantha for their valuable advice and comments to improve my work.

As well, I am grateful to all of the academic staff of the Faculty of Information Technology who took lectures in our postgraduate programme, and specially the course coordinator – Mr.S.C.Premaratne, without their teaching and guidance, I could not have got the capacity to successfully complete this project. Further, the course – Literature Review & Thesis Writing taught by Prof. ASK Karunanda - has particularly supported very much to document this research work successfully.

Furthermore, I would like to extend my sincere thanks to my sectional head at the office – Eng. D.S.D.Jayasiriwardena (Additional General Manager) for his encouragement and support to pursue the postgraduate programme successfully. Additionally, I would like to thankfully acknowledge the financial support of the National Water Supply and Drainage Board to cover the fees of my first year programme.

Special heartiest thanks should go to my family, Ms. Bhavani, Mas.Ashvin and Mas. Avaneesh, who always showed me their unconditional love, faith and support during this task. Moreover, I would like to thanks to my friends who encouraged to perform the project.

Moreover, I would like to express my gratitude to Park & Miller, Carta, Marsaglia, University of Sheffield, GlibC, ANSIC and National Security Agency for utilizing their proposed pseudo random number generators and secure hash algorithms in this study.

Finally, I humbly bow before the almighty God for showering his blessings upon me and giving me the strength, wisdom and luck to reach this important milestone in my academic life.

## Abstract

Water supply and sanitation services can no longer tolerate inefficiencies of their traditional non-intelligent distribution infrastructures due to the growth of demand for their uninterrupted services in quantity and quality wise. Wireless Sensor Networks can be employed to address these issues in a very cost effective manner. Already, Wireless Sensor Networks have been started to utilize in some countries for implementing their infrastructures of water supply and sanitation services as intelligent to provide better services and reduce financial losses. Ensuring efficient and secure data communication in Wireless Sensor Network is one of the major aspects in its wide range of applications. But, security solutions developed for traditional networks are not suitable for Wireless Sensor Networks due to its specific features. Many researches have been carried out to propose suitable efficient and secure lightweight protocols for Wireless Sensor Networks to improve their data communication.

In this project, an efficient and secure lightweight protocol has been proposed for Wireless Sensor Networks. Many literatures related to various security threats, security protocols and key management schemes of Wireless Sensor Networks have been critically reviewed at the beginning of the study. Literatures regarding the pseudo random number generators and hash algorithms relevant to these security architectures have also been critiqued to analyse the suitability of them.

In 2004, Park and Shin have proposed a lightweight protocol called Lightweight Security Protocol(LiSP). The salient feature of this protocol is the novel rekeying mechanism to tradeoff between security and resource consumption for large scale sensor networks. In 2006, Sun and coworkers have presented a lightweight security protocol with similar key management scheme of Park and Shin, but improved security mechanism by employing a pseudo random number generator - Linear Congruential Generator(LCG). In 2015, Jain and Ojha have identified that Park-Miller pseudo random number generator is better than Linear Congruential Generator for the lightweight security protocol. Further, in 2015, Ojha and Jain analysed some other pseudo random generators to evaluate the performances of the lightweight security protocol and concluded that Park-Miller pseudo random number generator is the most suitable one. But, these studies didn't consider Park-Miller's latest recommendation, or



other variations of the pseudo random number generators. Pseudo random number generators play vital role in the security and efficiency of the lightweight protocols. Moreover, It has been identified that Secure Hash Algorithm-1(SHA-1) employed in this protocol has similar effect as pseudo random number generator in the security and efficiency of the protocol.

Therefore, the performance of the lightweight protocol can be enhanced without compromising its security features, by utilizing more appropriate pseudo random number generator and hash function in its architecture.

So, the latter part of the project, the secure lightweight protocols having different pseudo random number generators and secure hash algorithms have been designed and implemented to evaluate their suitability for proposing an efficient and secure lightweight protocol.

Implementations have been modeled and evaluated in MATLAB software which had been recommended and utilized in many previous literatures for this purpose. Times taken for the computations have been analysed with pseudo random number generators and secure hash algorithms employed with their specific features.

The pseudo random number generator, LCG Sheffield, has been identified as a most suitable pseudo random number generator for the lightweight protocol. Secure Hash Algorithm -1 proposed in the previous studies has been identified as a most efficient hash function for the lightweight protocol.

This study proposes a secure lightweight protocol which is experimentally shown as, in average, 5.7% more efficient than the secure protocol proposed in the study by Jain and Ojha in 2015.

# Table of Contents

	Page No.
Abstract.....	v
List of Figures.....	xi
List of Tables.....	xiii
Abbreviations.....	xv
Chapter 1 – Introduction .....	1
1.1 Prolegomena .....	1
1.2 Background and Motivation.....	1
1.3 Problem definition.....	4
1.4 Aim and Objectives .....	4
1.4.1 Objectives related to the problem .....	5
1.4.2 Objectives related to the solution .....	5
1.5 Hypothesis .....	6
1.6 Resources required .....	6
1.7 Structure of the thesis .....	6
1.8 Summary .....	6
Chapter 2 – Developments in security of Wireless Sensor Network .....	7
2.1 Introduction .....	7
2.2 Review of Literature.....	7
2.3 Summarization of the reviews and Problem definition .....	12
2.3.1 Problem Definition .....	15
2.3.2 Identified Technologies .....	15
2.4 Summary .....	15
Chapter 3 – Technologies adopted for the proposed protocol.....	16
3.1 Introduction.....	16



	<b>Page No.</b>
3.2 Onetime pad(OTP) .....	16
3.3 Pseudo Random Number Generator (PRNG) .....	17
3.3.1 Linear Congruential Generator (LCG) .....	17
3.3.2 Park-Miller pseudo random number generator .....	18
3.3.3 Park-Miller-Carta pseudo random number generator .....	19
3.4 Block cipher .....	20
3.5 Dynamic key .....	20
3.6 Hash function .....	21
3.6.1 Secure Hash Algorithm -1 .....	21
3.7 Symmetric key cryptosystems .....	22
3.8 Encryption primitives of cryptosystems .....	22
3.8.1 Addition operation .....	23
3.8.2 Subtract operation .....	23
3.8.3 XOR operation .....	23
3.8.4 Transpose operation .....	24
3.8.5 Swap operation .....	24
3.9 MATLAB .....	24
3.10 Summary .....	25
<b>Chapter 4 – Approach for Efficient and Secure Lightweight Protocol .....</b>	<b>26</b>
4.1 Introduction .....	26
4.2 Hypothesis .....	26
4.3 Input.....	26
4.4 Output.....	27
4.5 Process.....	27
4.5.1 Encryption process.....	27
4.5.2 Decryption Process .....	28
4.6 Features.....	30
4.7 Users.....	31

	Page No.
4.8 Summary .....	31
Chapter 5 – Design of the Efficient and Secure Lightweight Protocol.....	32
5.1 Introduction .....	32
5.2 Top Level Architecture of the Secure Lightweight Protocol .....	32
5.2.1 Generation of Dynamic Binary Key .....	32
5.2.2 Encryption Process .....	34
5.2.3 Decryption Process .....	35
5.3 Theoretical Analysis of the Architecture .....	39
5.3.1 Selection of PRNGs .....	39
5.3.2 Selection of Hash Functions .....	41
5.4 Summary .....	42
Chapter 6 – Implementation of the Efficient and Secure Lightweight Protocol .....	43
6.1 Introduction.....	43
6.2 Overview of the implementation.....	43
6.3 Software and Platform used for the implementation.....	43
6.4 Implementation of the module ‘Generate Dynamic Binary Key’ .....	44
6.5 Implementation of the function ‘EncryptionProcess’ .....	46
6.6 Implementation of the function ‘DecryptionProcess’ .....	47
6.7 Implementation of the Module ‘AutomateDataCollection_PRNG’ .....	48
6.8 Implementation of the Module ‘AutomateDataCollection_HF’ .....	48
6.9 Implementation of the Module ‘AutomateFunctionalityTesting’ .....	49
6.10 Summary .....	49
Chapter 7 – Evaluation.....	50
7.1 Introduction .....	50
7.2 Testing the functionality of the Implementation .....	50
7.3 Evaluation Strategy .....	50

	Page No.
7.4 Average execution times of the Implementations with different PRNGs	51
7.5 Average execution times of the implementations with different Hash Algorithms.....	56
7.6 Summary .....	60
<b>Chapter 8 – Conclusion and Further Work .....</b>	<b>61</b>
8.1 Introduction .....	61
8.2 Conclusion .....	61
8.3 Further work .....	62
8.4 Summary .....	62
References .....	63
Appendix A – Detailed design Diagram .....	67
Appendix B – Selected Source Code .....	71
B.1 Listing ‘AutomateDataCollection_PRNGEvaluation’ (Only the specific modules) .....	71
B.2 Code Listing ‘AutomateDataCollection_HF’ (Only the specific modules) .....	93
Appendix C – Measured Execution Times.....	104
Appendix D – Results of the Functionality Testing .....	113
D.1 Evaluating architectures having different PRNGs .....	113
D.2 Evaluating architectures having different Hash Algorithms .....	115
D.3 Results of the Automated Functionality Testing.....	118
Appendix E – Screen Images .....	119
E.1 Testing the Functionalities of the Implemented Designs .....	119
E.2 Demonstrating the Processes of the Architecture.....	120





# List of Figures

No.	Description No.	Page
Figure 1.1	Some Smart components in a Smart City	2
Figure 1.2	Some applications of WSNs in the infrastructure of a Water Supply System	3
Figure 3.1	Addition operation	23
Figure 3.2	Subtraction operation	23
Figure 3.3	XOR operation	23
Figure 3.4	Transpose operation	24
Figure 3.5	Swap operation	24
Figure 5.1	Top level architecture of the Efficient and Secure Lightweight Protocol	33
Figure 5.2	Architecture of the Module 'Generate Dynamic Binary Key'	34
Figure 5.3	Architecture of the Module 'Encrypt Plain Data'	37
Figure 5.4	Architecture of the Module 'Decrypt Cipher Data'	38
Figure 7.1	Execution time vs PRNGs for the input data size 25Kbyte	52
Figure 7.2	Execution time vs PRNGs for the input data size 30Kbyte	52
Figure 7.3	Execution time vs PRNGs for the input data size 35Kbyte	53
Figure 7.4	Execution time vs PRNGs for the input data size 40Kbyte	53
Figure 7.5	Execution time vs PRNGs for the input data size 45Kbyte	54
Figure 7.6	Execution time vs PRNGs for the input data size 50Kbyte	54
Figure 7.7	Execution time vs PRNGs for the input data size 55Kbyte	55
Figure 7.8	Execution times of LCGSheffield vs ParkMiller PRNGs architectures	56
Figure 7.9	Execution time vs Hash Algorithms for the input data size 25Kbyte	58
Figure 7.10	Execution time vs Hash Algorithms for the input data size 30Kbyte	58
Figure 7.11	Execution time vs Hash Algorithms for the input data size 35Kbyte	59

Figure 7.12	Execution time vs Hash Algorithms for the input data size 40Kbyte	59
Figure 7.13	Execution time vs Hash Algorithms for the input data size 45Kbyte	60
Figure 7.14	Execution time vs Hash Algorithms for the input data size 50Kbyte	60
Figure 7.15	Execution time vs Hash Algorithms for the input data size 55Kbyte	61
Figure A.1	Architecture of the Module 'Encrypt Data Block: Round 1'	68
Figure A.2	Architecture of the Module 'Encrypt Data Block: Round 2'	68
Figure A.3	Architecture of the Module 'Encrypt Data Block: Round 3'	69
Figure A.4	Architecture of the Module 'Encrypt Data Block: Round 4'	69
Figure A.5	Architecture of the Module 'Decrypt Data Block: Round 1'	70
Figure A.6	Architecture of the Module 'Decrypt Data Block: Round 2'	70
Figure A.7	Architecture of the Module 'Decrypt Data Block: Round 3'	71
Figure A.8	Architecture of the Module 'Decrypt Data Block: Round 4'	71
Figure E.1	Successive screen shots of testing the functionality of the designs with different PRNGs	122
Figure E.2	Successive screen shots of the demonstration processes of the architecture.	127

## List of Tables

No.	Description	Page No.
Table 2.1	Summarization of the identified issues in the literature review.	10
Table 5.1	Values of the constants of well-accepted PRNGs	38
Table 5.2	Values of the constants of Selected PRNGs	39
Table 5.3	Properties of the selected Hash Algorithms	
Table 7.1	Measured average execution times for different sizes of input data, with the same hash algorithm.	49
Table 7.2	Efficiency comparison architectures having LCGSheffield and ParkMillereen	54
Table 7.3	Measured average execution times for different sizes of input data, with the same PRNG.	55
Table C.1	Measured execution times of the input data 25 Kbyte - same PRNG	114
Table C.2	Measured execution times of the input data 30 Kbyte - same PRNG	115
Table C.3	Measured execution times of the input data 35 Kbyte - same PRNG	115
Table C.4	Measured execution times of the input data 40 Kbyte - same PRNG	116
Table C.5	Measured execution times of the input data 45 Kbyte - same PRNG	117
Table C.6	Measured execution times of the input data 50Kbyte - same PRNG.	117
Table C.7	Measured execution times of the input data 55Kbyte - same PRNG.	118
Table C.8	Measured execution times of the input data 25Kbyte - same Hash Algorithm	119
Table C.9	Measured execution times of the input data 30Kbyte - same Hash Algorithm.	119



Table C.10	Measured execution times of the input data 35Kbyte - same Hash Algorithm.	120
Table C.11	Measured execution times of the input data 40Kbyte - same Hash Algorithm.	121
Table C.12	Measured execution times of the input data 45Kbyte - same Hash Algorithm.	121
Table C.13	Measured execution times of the input data 50Kbyte - same Hash Algorithm.	122
Table C.14	Measured execution times of the input data 55Kbyte - same Hash Algorithm.	122
Table D.1	List of cipher texts and decrypted plain texts generated by the protocol which have different PRNGs in its architecture	125
Table D.2	List of cipher texts and decrypted plain texts generated by the protocols which have different the Hash Algorithms in its architecture	127
Table D.3	Test results of the Automated Functionality Testing	128

# Abbreviations

$\mu$ TESLA	Micro version of Timed Efficient Streamed Loss-tolerant Authentication
ANSI	American National Standards Institute
IDS	Intrusion Detection System
IoT	Internet of Things
LCG	Linear Congruential Generator
LiSP	Lightweight Security Protocol
LLSP	Link Layer Security Protocol
LSec	Lightweight Security Protocol
MAC	Message Authentication Code
MEMS	Micro Electro Mechanical Systems
NIST	National Institute of Standards and Technology
NSA	National Security Agency
PRNG	Pseudo Random Number Generator
SHA-1	Secure Hash Algorithm -1
SNEP	Sensor Network Encryption Protocol
WSN	Wireless Sensor Network