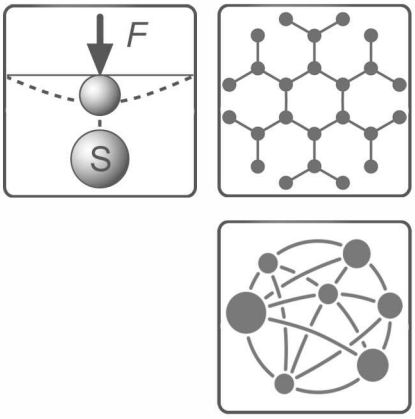
ADVANCES IN SENSORS: REVIEWS 7

Sergey Y. Yurish
Editor

Physical and Chemical Sensors: Design, Applications & Networks



IFSA

Sergey Y. Yurish
Editor

Physical and Chemical Sensors: Design, Applications & Networks

Book Series: Advances in Sensors: Reviews, Vol. 7

International Frequency Sensor Association Publishing

Sergey Y. Yurish
Editor

Physical and Chemical Sensors: Design, Applications & Networks
Advances in Sensors: Reviews, Vol. 7

Published by International Frequency Sensor Association (IFSA) Publishing, S. L., 2019
E-mail (for print book orders and customer service enquires): ifsa.books@sensorsportal.com

Visit our Home Page on http://www.sensorsportal.com

Advances in Sensors: Reviews, Vol. 7 is an open access book which means that all content is freely available without charge to the user or his/her institution. Users are allowed to read, download, copy, distribute, print, search, or link to the full texts of the articles, or use them for any other lawful purpose, without asking prior permission from the publisher or the authors. This is in accordance with the BOAI definition of open access.

Neither the authors nor International Frequency Sensor Association Publishing accept any responsibility or liability for loss or damage occasioned to any person or property through using the material, instructions, methods or ideas contained herein, or acting or refraining from acting as a result of such use.

ISBN: 978-84-09-14512-6
e-ISBN: 978-84-09-14511-9
BN-20191230-XX
BIC: TJFC



Contents

Contents **7**
Contributors **13**
Preface **17**

Chapter 1
CMOS Readout of MEMS Flow Sensors **19**
1.1. Introduction *19*
1.2. Flow Sensors *20*
1.2.1. Mass Flow Based Flow Sensors 20
1.2.1.1. Coriolis Flow Meters 20
1.2.1.2. Thermal Mass Flow Meters 21
1.2.2. Velocity Based Flow Sensor 21
1.2.2.1. Ultrasonic Flow Meters 22
1.2.2.2. Vortex Flow Meters 22
1.2.3. Differential Pressure Based Flow Sensors 22
1.2.3.1. Orifice Plate Flow Meter 22
1.2.3.2. Venturi Tube Flow Meter 23
1.2.4. Positive Displacement Flow Sensors 23
1.3. MEMS Flow Sensor for Our Application *24*
1.3.1. Proposed MEMS Flow Sensor 24
1.3.2. Heater Control Circuit 25
1.4. CMOS Readout Circuit for MEMS Flow Sensor *27*
1.4.1. First Generation CMOS MEMS Flow Sensor 27
1.4.1.1. Design Considerations for the CMOS Readout Circuit 28
1.4.1.2. Transistor-level Implementation of CFIA 30
1.4.1.3. Measurement Results 32
1.4.2. Second Generation CMOS MEMS Flow Sensor 35
1.4.2.1. Readout Circuit Design of 2nd Gen CMOS MEMS Flow Sensor 35
1.4.2.2. Measurement Results of 2nd Gen CMOS MEMS Flow Sensor 39
1.5. Conclusion *44*
References *45*

Chapter 2
Vehicle's Axle Load Sensors in Weigh-in-motion Systems **49**
2.1. Background *49*
2.2. Principle of Axle Load Sensors *52*
2.3. Types of Axle Load Sensors *55*
2.3.1. Polymer Sensors 56
2.3.2. Quartz Sensors 57
2.3.3. Strain Gauge Sensors 59
2.3.4. Capacitive Sensors 61
2.3.5. Fiber Optic Sensors 62
2.4. Comparative Study *63*
2.5. Summary *65*
References *66*

Chapter 3
Integrated-optics Resonator (IORG) Gyros: A Review 69
3.1. Introduction 69
3.2. IORG: Main Configurations Looking for the Progressive Reduction of Parasitic Optical Noises 74
3.2.1. Research Results Obtained by H. Ma *et al.* Group 74
3.2.2. Research Results Obtained by L. Feng *et al.* Group 79
3.3. IORG: Hybrid and Monolithic Integration Solutions 83
3.4. IORG: Advanced Design, Trends and Optimization 87
3.5. Conclusions 89
Acknowledgements 89
References 89

Chapter 4
Ionic Liquids in Chemical Sensing: A Window of Opportunities 93
Abbreviations 93
4.1. Introduction 95
4.2. Ionic Liquids as Chemical Sensors 96
4.2.1. Heavy Metal Sensing 97
4.2.2. Ion Sensing 100
4.2.3. Organic Compounds Sensing 102
4.2.4. Gas Sensing 105
4.2.4.1. Oxygen (O_2) Gas Sensing 106
4.2.4.2. Carbon Dioxide (CO_2) Sensing 107
4.2.4.3. Hydrogen (H_2) Gas Sensing 108
4.2.4.4. Ammonia (NH_3) Gas Sensing 110
4.2.4.5. Volatile Organic Compounds Sensing 111
4.2.5. Bioanalyte Sensors 113
4.3. Conclusions 116
Acknowledgments 117
References 117

Chapter 5
High Sensitively Chemiresistive Sensor Based on Novel 2D MXene and TiO_2 Nanocomposite for Detection of Anti-cancer Agent, 8-HOA 125
5.1. Introduction 125
5.2. Experiments 127
5.2.1. MXenes Synthesis and Characteristics 127
5.2.1.1 TiO_2 NWs Synthesis 127
5.2.1.2. MXenes Ti_3C_2 Synthesis 127
5.2.1.3. Ti_3C_2 MXene& TiO_2 NWs synthesis 128
5.2.2. Real-time Sensing Detection of Variable Concentration of 8-HOA 128
5.3. Discussion 129
5.3.1. Material Characterization 130
5.3.2. 8-HOA Sensing Response Tests 133
5.4. Conclusion 134
Acknowledgements 135
References 135

Chapter 6
Humidity Measurement in Gases 139
6.1. Humidity Measurement Methods 139

6.2. Thermodynamic Terms in Humidity Measurement *139*
6.3. Overview of the Most Commonly Used Humidity Measurement Devices *141*
6.4. Principles of Humidity Measurement in Gases *141*
6.4.1. Dew Point Hygrometer 141
6.4.2. Psychrometer 145
6.4.3. Capacitive Polymer Sensors 146
6.4.4. Coulometric Measurement 148
6.4.5. Metal Oxide Sensors 149
6.4.6. Measurement of the Optical Properties of Water and Water Vapor 151
6.4.7. Hair and Fiber Hygrometer 153
6.4.8. Zirconium Oxide Devices 154
6.4.9. Quartz Microbalance 155
6.4.10. Measurement of Acoustic Properties of Water Vapor 156
6.4.11. Resistive Sensors 157
6.4.12. Humidity Indicator 158
6.4.13. Nanostructured Measurement Devices 160
6.4.13.1. Contact Methods 160
6.4.13.2. Noncontact Methods 161
References *162*

Chapter 7
RGB Multispectral Microanalyzer 165
7.1. Introduction *165*
7.2. Microanalizer Development *166*
7.2.1. Microfluidic Structure 167
7.2.2. Hydrodynamic 168
7.2.3. RGB Optical Detection System 169
7.2.3.1. Radiation Resource Calibration 169
7.2.3.2. The radiation Flow Detection 171
7.2.4. Electronic Control and Wireless for Automation Process 171
7.3. Analytical *171*
7.4. Experimental Set-up *172*
7.5. Results and Discussion *173*
7.6. Conclusion *176*
Acknowledgements *176*
References *176*

Chapter 8
Energy Audit and Contactless Temperature Measurement 179
8.1. Introduction *179*
8.1.1. Reconstruction of Buildings 179
8.1.2. Energy Audit 182
8.1.3. Replacement of Energy-inefficient Heating Systems 182
8.1.4. Construction of Smart Energy-efficient Buildings 182
8.2. Improvement of Energy Balance of the House due to Exact Contactless Temperature Measurement of Surfaces *184*
8.3. The practice of Pyrometry and Analysis of Problem *188*
8.4. Minimization of the Measurement Error of Pyrometer *192*
8.5. Example of Implementing the Method *194*
8.6. Conclusions *195*
Acknowledgment *195*
References *195*

Chapter 9
Indoor Stereo Photogrammetry via Omnidirectional Multicamera System Case Study: Ladybug2 197
9.1. Introduction 197
9.2. Coordinate Systems and Calibration 202
9.3. Application 209
9.4. Results of Application 215
9.5. Conclusion 220
References 222

Chapter 10
Improved PWM A/D Conversion Technique: Working Principle and Model Validation 225
Abbreviations and Symbols 225
10.1. Introduction 226
10.2. Basics of PWM A/D Conversion 227
10.2.1. Traditional PWM Based A/D Conversion 227
10.3. Improved PWM Based A/D Conversion 231
10.3.1. Block Diagram and Principle of Operation 231
10.3.2. Expression of the Converter Characteristic Function 232
10.3.3. RC Time Constant Dimensioning 233
10.3.4. Conversion Range 235
10.4. Simulation Results 236
10.4.1. Voltage Variation in the Capacitor during the Conversion Cycle 236
10.4.2. Quantization Errors for a Sinusoidal Signal 237
10.5. Experimental Results 241
10.6. Conclusions 243
References 244

Chapter 11
Improved PWM A/D Conversion Technique: Applications for Digital Sensors and Sensor Systems 247
Abbreviations and Symbols 247
11.1. Introduction 247
11.2. Main Advantages of PWM NPC A/D Conversion Technique 248
11.2.1. Linearization of Characteristics 249
11.2.2. Conversion Range Adjustment 253
11.3. Auto-calibration 253
11.4. Experimental Study Cases 258
11.4.1. Linearization of a Thermistor Characteristic 258
11.4.2. Linearization a Flow Meter Characteristic 260
11.5. Conclusions 263
References 263

Chapter 12
Energy Efficient Key Management Protocol for Securing Autoconfigurable Networks 265
12.1. Introduction 265
12.2. Background and Related Works 266
12.2.1. Symmetric Key Cryptographic Systems 267

12.2.1.1. Random Key Pre-distribution Schemes 268
12.2.1.2. Schemes Based on the Base Station Participation or on a Master Key Pre-distribution 271
12.2.2. Asymmetric Key Cryptographic Systems 273
12.2.2.1. PKI Based Schemes 273
12.2.2.2. Node Identity Based Schemes 274
12.2.3. Comparative Analysis 275
12.3. LESDTP: A New Key Management Protocol 277
12.3.1. Protocol Overview 277
12.3.2. Assumptions 278
12.3.3. Notations 278
12.3.4. Phases of the LESDTP Protocol 279
12.3.4.1. Key Management and Network Implementation Phase 279
12.3.4.2. Key Maintenance Phase 284
12.4. Evaluation of LESDTP Protocol 284
12.4.1. Security Services Offered by the LESDTP Protocol 285
12.4.1.1. Data Integrity and Source Node Authentication 285
12.4.1.2. Data Confidentiality 285
12.4.1.3. Data Freshness 285
12.4.2. LESDTP's Resistance to Attacks 286
12.4.3. LESDTP Performance Analysis 286
12.4.3.1. Attacker Model 286
12.4.3.2. LESDTP Analysis Relatively to This Attacker Model 286
12.4.4. LESDTP Comparison with Some Protocols of the State of the Art 287
12.5. Conclusion and Future Works 287
References 288

Index 291

Contributors

Moaaz Ahmed Division of Information and Computing Technologies, College of Science and Engineering, Hamad Bin Khalifa University, Education City, Qatar

O. Ama Department of Electrical and Computer Engineering, North Dakota State University, Fargo, ND 58108, USA
Biomedical Engineering Program, North Dakota State University, Fargo, ND 58108, USA

Kamalakanta Behera Centre for Interdisciplinary Research in Basic Sciences, JMI, Jamia Nagar, New Delhi, 110025, India.
Department of Chemistry, Indian Institute of Technology Delhi, Hauz Khas, New Delhi, 110016, India, E-mail: kk.chemistry.iitd@gmail.com

Amine Bermak Division of Information and Computing Technologies, College of Science and Engineering, Hamad Bin Khalifa University, Education City, Qatar

Bhawna Department of Chemistry, Indian Institute of Technology Delhi, Hauz Khas, New Delhi, 110016, India

P. Burnos AGH University of Science and Technology, Department of Measurement and Electronics, 30 Mickiewicz Avenue, 30-059 Krakow, Poland

Juana M. Camarillo Electric-Electronic Departament. National Technology of Mexico – La Laguna Institute of Technology, Torreón, Coahuila, Mexico

Rosa M. Camarillo Mechanic and Mechatronic Departament, National Technology of Mexico – La Laguna Institute of Technology, Torreón, Coahuila, Mexico

Divya Dhingra Department of Chemistry, Indian Institute of Technology Delhi, Hauz Khas, New Delhi, 110016, India

Chérif Diallo Laboratoire Algèbre, Cryptographie, Codes et Applications (LACCA), UFR Sciences Appliquées et de Technologies (UFR SAT), Université Gaston Berger, Saint-Louis, Sénégal

José Miguel Costa Dias Pereira ESTSetúbal/IPS, Setúbal, Portugal
Instituto de Telecomunicacões, Lisboa, Portugal

Bahadir Ergun Gebze Technical University, Department of Geomatics Engineering, 41400 Gebze, Kocaeli, Turkey

José Flores CUCEI (University Center of Exact Sciences and Engineering) Guadalajara University, Guadalajara, Jalisco, México

J. Gajda AGH University of Science and Technology, Department of Measurement and Electronics, 30 Mickiewicz Avenue, 30-059 Krakow, Poland

Alejandra Guerrero Mechanic and Mechatronic Departament, National Technology of Mexico – La Laguna Institute of Technology, Torreón, Coah, Mexico

M. Johnson Department of Electrical and Computer Engineering, North Dakota State University, Fargo, ND 58108, USA
Materials and Nanotechnology Program, North Dakota State University, Fargo, ND 58108, USA

Anu Kadyan Department of Chemistry, Indian Institute of Technology Delhi, Hauz Khas, New Delhi, 110016, India

Lizhi Liu Department of Pharmaceutical Science, North Dakota State University, Fargo, ND 58108, USA

Mykola Mykyychuk Lviv Polytechnic National University, Bandery str., 12, 79013, Lviv, Ukraine

Siddharth Pandey Department of Chemistry, Indian Institute of Technology Delhi, Hauz Khas, New Delhi, 110016, India

Ramón José Pérez Menéndez UNED-Spain, Barrio Tolda de Castela, 4, 27002, Lugo, Spain, Tel: +34 982 280 202, fax: +34 982 231 600,
E-mail: ramonjose.perez@lugo.uned.es

Steven Y. Qian Department of Pharmaceutical Science, North Dakota State University, Fargo, ND 58108, USA

Cumhur Sahin Gebze Technical University, Department of Geomatics Engineering, 41400 Gebze, Kocaeli, Turkey

Pylyp Skoropad Lviv Polytechnic National University, Bandery str., 12, 79013, Lviv, Ukraine

R. Sroka AGH University of Science and Technology, Department of Measurement and Electronics, 30 Mickiewicz Avenue, 30-059 Krakow, Poland

Bohdan Stadnyk Lviv Polytechnic National University, Bandery str., 12, 79013, Lviv, Ukraine

Shruti Trivedi Department of Chemistry, Indian Institute of Technology Delhi, Hauz Khas, New Delhi, 110016, India

Arturo Urquizo Mechanic and Mechatronic Departament, National Technology of Mexico – La Laguna Institute of Technology, Torreón, Coahuila, Mexico

José Valderrama Electric-Electronic Departament. National Technology of Mexico – Nuevo León Institute of Technology, Monterrey, Nuevo León, Mexico

D. L. Wang Department of Electrical and Computer Engineering, North Dakota State University, Fargo, ND 58108, USA, Biomedical Engineering Program, North Dakota State University, Fargo, ND 58108, USA, Materials and Nanotechnology Program, North Dakota State University, Fargo, ND 58108, USA, E-mail: danling.wang@ndsu.edu, Tel: 86-10-701-231-8396

Jan Wernecke Feuchtemesstechnik GmbH, Germany

Roland Wernecke Feuchtemesstechnik GmbH, Germany

Anita Yadav Department of Chemistry, Indian Institute of Technology Delhi, Hauz Khas, New Delhi, 110016, India

Svyatoslav Yatsyshyn Lviv Polytechnic National University, Bandery str., 12, 79013, Lviv, Ukraine, E-mail: slav.yat@gmail.com

Q. Zhang Department of Electrical and Computer Engineering, North Dakota State University, Fargo, ND 58108, USA, Biomedical Engineering Program, North Dakota State University, Fargo, ND 58108, USA, Materials and Nanotechnology Program, North Dakota State University, Fargo, ND 58108, USA

Preface

I am very happy to present the 7th volume of the popular '*Advances in Sensors: Reviews*' Book Series started by the IFSA Publishing in 2012. Today this open access Book Series is a primary sensor review source published for the sensor community.

The very rapid progress in sensor research, applications and technologies are continuing during the last decade. Sensors have become one of the most important and widely used components in various applications. According to Allied Market Research (AMR), the global market of sensors and main types of transducers is poised to grow with a compound annual growth rate of 11.3 % until 2022 when the market will reach US$ 241 billion. The increasing utilization and penetration of Internet of Things (IoT), coupled with increasing automation in vehicles and smart wearable systems for health monitoring, is expected to augment the growth of the sensor market over the forecast period.

Today, more and more sensor manufacturers are using machine learning to sensors and signal data for analyses. The machine learning and signal data analyses are becoming easier than ever: hardware is becoming smaller and sensors are getting cheaper, making Internet of things devices widely available for a variety of applications ranging from predictive maintenance to user behavior monitoring. To guarantee the steady progress in sensor manufacturing and development, the strong connection of sensor industry with the academy is necessary. The '*Advances in Sensors: Reviews*' Book Series is a significant tool for such relationship.

The 7th volume entitled '*Physical and Chemical Sensors: Design, Applications & Networks*' contains 12 chapters with sensor related state-of-the-art reviews and descriptions of latest achievements written by 34 authors-experts from academia and industry from 12 countries: Germany, Hong Kong, India, Mexico, Poland, Portugal, Qatar, Senegal, Spain, Turkey, Ukraine and USA. It is my great pleasure to see International Frequency Sensor Association (IFSA) members and active participants of the annual International Conference on Sensors and Electronic Instrumentation Advances (SEIA) among the authors.

The coverage includes current developments in physical sensors (MEMS flow sensors, load sensors and gyroscopes); chemical sensors for humidity in gases and anti-cancer agents; readout circuitry for various PWM output sensors and transducers; and autoconfigurable wireless sensor networks.

This book ensures that our readers will stay at the cutting edge of the field and get the right and effective start point and road map for the further researches and developments. By this way, they will be able to save more time for productive research activity and eliminate routine work.

With this unique combination of information in each volume, the *'Advances in Sensors: Reviews'* book Series will be of value for scientists and engineers in industry and at universities, to sensors developers, distributors, and users.

Like the first six volumes of this book Series, the 7th volume also has been organized by topics of high interest. In order to offer a fast and easy reading of the state of the art of each topic, every chapter in this book is independent and self-contained. All chapters have the same structure: first an introduction to specific topic under study; second particular field description including sensing applications. Each of chapter is ending by well selected list of references with books, journals, conference proceedings and web sites.

The new book volume is also published in two formats: electronic (pdf) with full-color illustrations and print (paperback). I shall gratefully receive any advices, comments, suggestions and notes from readers to make the next volumes of 'Advances in Sensors: Reviews' book Series very interesting and useful.

Dr. Sergey Y. Yurish

Editor
IFSA Publishing *Barcelona, Spain*

Chapter 12
Energy Efficient Key Management Protocol for Securing Autoconfigurable Networks

Chérif Diallo

12.1. Introduction

Security issues related to communication in autoconfigurable networks such Wireless Sensor Networks (WSN) are numerous and complex [1-4]. This complexity is most often linked to the constraints conveyed by the sensors [5]. Indeed, the sensors are characterized as entities that have low storage memory, low computing power and the battery that it integrates is in no way rechargeable [5]. Because of its limitations, traditional cryptographic solutions like RSA that require a lot of calculations in the establishment, encryption and decryption of keys become inapplicable for this type of network [6]. This means that lightweight cryptography [7-10], which could be an alternative to the old cryptographic algorithms, is aimed to apply security mechanisms like cryptographic algorithms in WSN while optimizing the limited resources of the sensors. Optimization will concern, among others, storage, memory, bandwidth, energy. Then lightweight cryptography mechanisms are intended to be able to authenticate data transmission as well as a source of diffusion, to ensure data integrity, confidentiality, and availability. Moreover, managing scalability of autoconfigurable networks is an important issue because sometimes it may be necessary to add more nodes while also ensuring the freshness of data [1, 2] and the quality of service in the network.

Nevertheless, many attacks [1, 2, 11-14] still remain in these networks because most of the proposed solutions are often limited to a very particular operation (clustering xor routing, xor data transmission) and do not take into account all operations at the same time.

Cherif Diallo
Laboratoire Algèbre, Cryptographie, Codes et Applications (LACCA), UFR Sciences Appliquées et de Technologies (UFR SAT), Université Gaston Berger, Saint-Louis, Sénégal

With all these considerations, we propose in this chapter, the Low Energy Secure Data Transmission Protocol (LESDTP) [15] which is a new key management protocol for overall clustering process and data transmission operations in autoconfigurable wireless networks such as WSN.

12.2. Background and Related Works

Many cryptography solutions (encryption algorithms, authentication and key management methods, etc.) have been designed for wired networks but these solutions dot work well in some wireless networks like Mobile Adhoc Networks (MANETs) or Wireless Sensor Networks (WSNs). The most frequently proposed solutions use methods and standards in such a way that does not completely meet the needs of the WSN, because of the limited node capacity in energy resources, memory and computation.

The adaptation of any cryptography mechanism to the WSNs must comply with the following conditions: (i) the exchange of keys and information concerning their creation must be carried out rapidly and securely since the communications pass via the radio signals and could be intercepted at any time, (ii) the nature of the WSN deployment for some applications makes the nodes vulnerable to physical capture (the loss of these nodes should not affect the security of communications across the entire network), (iii) WSN nodes are known to be limited in energy resources, computing potency and storage capacity. Energy saving in WSN is a big issue for any application. Thus, the architecture proposed for a given application must find a compromise between the level of security and the total cost of resource consumption. For example, the use of a single encryption key for the entire network could help in reducing the required memory quantity, and eliminates the calculation of derived keys and thus allows energy saving. However, the use of single encryption solutions does not offer a good level of security over time. An opponent could extract the key from a physically captured node and decipher information that circulates in the network. Hence the need for certain applications of having a sufficiently secure architecture while respecting the weak capacity of the nodes.

The establishment of a secret key between two or more nodes is one of the most important security services that ensures the confidentiality and integrity in a WSN. In order to achieve this goal in a secure manner, we need a key management protocol that can manage: (i) pre-distribution of keys prior to network deployment, (ii) establishment of keys in a secure way after deployment, and (iii) the issuance, renewal and revocation of keys throughout the network lifetime. The dynamic character of key management will allow the network to remain scalable and to be able to manage the coming arrival of nodes in the WSN or the node losses. Moreover, this must not only ensure quick node integration by delivering certified keys, but must also provide keys revokation or renewal for maintenance services.

The easiest way to encrypt/decrypt messages exchanged between nodes relies on symmetric key cryptographic systems. For sensitive applications, the symmetric key could be renewed periodically to deal with some threats like eavesdropping. However, if an

attacker succeeds in recovering this key by calculating it or by capturing a node, he would be able to decipher messages and compromise the network security.

Other methods are based on the mechanisms of asymmetric key cryptographic systems including the concept of Public Key Infrastructure (PKI) derived from wired networks. For example, some methods use the base station as a trusted third party to replace the PKI CA (Certificate Authority). In this way, the base station will be responsible for authenticating and distributing authentication keys after network deployment. The difficulty of such cryptographic methods is to take into account network bandwidth limitations, and energy, memory and node storage.

Depending on the applications, we could meet the use of symmetric cryptographic systems, or asymmetric cryptographic systems or hybrid systems which jointly use the symmetric and asymmetric systems (Fig. 12.1).

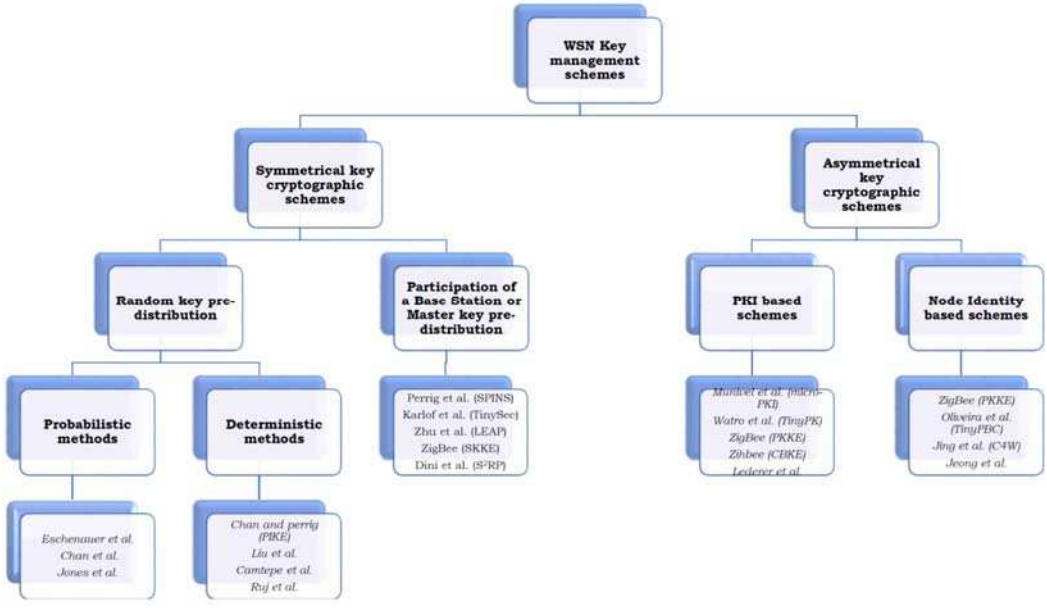


Fig. 12.1. A classification of WSN key management schemes.

There are other classification methods, some of which are based on the type of approach such as sharing a key between two nodes (Pair-wise) or between multiple nodes (Group-wise), and others, on the fact that they exploit the probabilities, combinatorial analysis, etc.

12.2.1. Symmetric Key Cryptographic Systems

Symmetric key cryptographic schemes use a common key for both encryption and decryption processes between two WSN nodes. This is done in three steps:

- **Key pre-distribution**: the set of keys stored in the node memory before network deployment are the key ring of the node. If there is a common key (or Shared-key) between two nodes, then they can create a secure link between them.
- **Shared-key discovery**: after network deployment, a communication protocol will be responsible for the common key discovery process between two neighboring nodes.
- **Path-key establishment**: if there is no common key between two nodes wanting to communicate, then we must find a secure path between them. This path goes through a set of nodes that already contains secure links. Once this path is established, both nodes can use it to communicate securely.

12.2.1.1. Random Key Pre-distribution Schemes

Random key pre-distribution methods include: (i) Probabilistic methods, defined by a random selection of key sets selected from a large list knowing that there are common keys between its keychains, with some probability; (ii) Deterministic methods, in this case the keys sets are generated in a deterministic way to ensure the establishment of some links between the nodes, and (iii) The hybrid methods, born from a combination of the two previous methods to increase the connectivity between nodes and enhance network scalability.

12.2.1.1.1. Random Key Pre-distribution with Probabilistic Methods

Eschenauer et al. [16] introduce the idea of key pre-distribution, before network deployment. For each node, a subset of keys is randomly chosen from a large set of keys. This is done in such a way that two neighboring nodes could have at least one key common with some probability after deployment. This method is considered as the basic scheme of probabilistic methods. The key pre-distribution phase of the scheme is carried out in several stages:

- It begins by generating a large set $\boldsymbol{P}$ of keys, pool of keys, with their identifiers.
- Then, for each node $\boldsymbol{i}$, a subset $\boldsymbol{k_i}$ of keys is randomly selected from $\boldsymbol{P}$. The keyring $\boldsymbol{k_i}$ is then stored in the memory of the node $\boldsymbol{i}$. The association list between the identifiers of the keyrings and the identifiers of the nodes is stored in a control node. A small number of keys stored in the nodes will permit establishment of common key between two nodes after network deployment.
- The second phase of this method is the discovery process of common keys. In order to discover if a neighboring node shares a common key, a source node broadcasts the list of identifiers of its keychain in plain text to its neighbors. Even if the opponent discovers this list, he will not have the opportunity to discover the keys of the list and he will not have the shared key used for encryption. However, if he captures the node, he will be able to discover the shared key and then decipher the communications of the established link.

- When a node is captured, its keychain must be revoked. A control node (which may be mobile) broadcasts a signed revocation message containing the list of identifiers of the revoked keychain. In order to sign this list, the control node generates a signature key, which is encrypted with the shared key between all nodes and the control node, and then sends it individually (unicast message) to the nodes. The nodes are then able to check the signature of the received message. Finally, each node locates the common identifiers with this list and then removes the corresponding keys from their memory.
- Once a keychain is revoked, some created links could disappear and then affect communications between nodes. But, affected nodes can reconfigure missing communication links by restarting a new key establishment phase with their neighbors. The key revocation only affects a few nodes and only a small portion of keys in their keychain will be removed. This is necessary to limit the risks associated with node capture attack.

In this scheme, the renewal of keys of a node is equivalent to a self-revocation of the shared key with another node. The node resumes the discovery phase of the common key with its neighbor without involving any control node in the operation. Note that after each renewal or revocation operation, a node loses at least one key to its keychain and may have some affected communication links with other nodes, or no communication link in case of exhaustion of the entire keychain.

Chan et al. [17] was inspired by *Eschenauer et al.* [16] method to propose a new pre-distribution scheme where two neighboring nodes must share several common keys instead of one. This increases the resistance against the node capture attack because the adversary will have more difficult to find the key established with an adjacent node among the common keys.

12.2.1.1.2. Deterministic Keys Pre-distribution Schemes

These methods have the advantage of guaranteeing fully connected graphs because they allow each node to establish a unique key with any other node in the network. However, this requires a large storage capacity. Known as pairwise key, each distributed key is unique and is established between only two nodes. There are several schemes for distributing symmetric keys in a deterministic way. We can classify them into three categories: grid-based schemes [18, 19], methods based on combinatorial structures [20, 21], and methods based on the use of codes [22].

Chan and Perrig [18] proposed a grid-based method called PIKE (Peer Intermediaries for Key Establishment). Most probabilistic schemes realize their keys distribution by storing the keys according to a complexity process of order ***O(N)*** (where ***N*** is the maximum number of nodes in the network). In addition, they often use the base station as a trusted intermediate node. This results in an energy overconsumption as this requires that many messages are sent towards the base station.

In order to solve the problems related to the storage of a large number of keys and the depletion of energy, PIKE proposes a deterministic key establishment method that can be realized according to a complexity process of order $O(\sqrt{N})$. This method uses ordinary network nodes as trusted intermediaries. The keys are pre-distributed so that any pair of nodes A and B can always have the possibility of finding a node C of the network that shares a unique key with A and B. Then Will use C as a trusted intermediate node to transmit the key establishment message to B.

In order to better understand how the method works, we use an example provided by the authors of PIKE [18]. The Fig. 12.2 shows a logical grid (or virtual space) of identifiers of 100 nodes. Each number in the grid is the identifier of a network node. This shows that the nodes 91 and 14 can share a unique key with each of the nodes that belong to their column or their line. Thus, the nodes 11 and 94 can act as trusted intermediaries in establishing keys between 91 and 14.

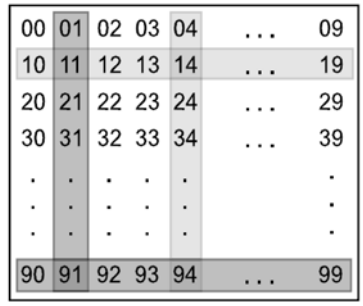


Fig. 12.2. Virtual space of node identifiers in a network of 100 nodes [18].

Each node in Fig. 12.2 will be loaded with 18 keys (9 keys for the nodes belonging to its line and 9 keys for the nodes belonging to its column). To generalize, each node of a network of size N is loaded with $2(\sqrt{N}-1)$ distinct keys. Once the trusted intermediary C has been chosen (geographically closer to A and B), A encrypts a new symmetric key $k_{A,B}$ to be shared with B using its common key with C and sends it to C. At the reception, C decrypts the message and retrieves the key $k_{A,B}$. Then, C re-encrypts it with the common key shared with B before sending it to B. Finally, B receives the shared key $k_{A,B}$ with A by decrypting the message received from C using the unique key shared with C. Finally, the node B sends an acknowledgment to the node A. Each message is accompanied by a MAC (message authentication code) to check its integrity.

Note that PIKE minimizes key storage in node memory before network deployment compared to other methods. However, the number of messages exchanged during the key establishment process has the disadvantage of consuming more energy and time. PIKE proposed the use of network nodes as trusted intermediaries in place of the base station. This has the advantage to avoid exhaustion battery of the nodes located near the base station. However, this solution could have a disadvantage, because any threat against the

trusted intermediate nodes of ***A*** and ***B*** could destroy the security of messages exchanged between ***A*** and ***B***.

Liu et al. [19] has proposed a similar grid based approach to *Chan and Perrig* [18] where the main idea is based on the location of the nodes grouping them in ***n*** groups each containing a number ***m*** of nodes. The pre-distribution scheme establishing a unique shared key between two nodes of the same group is called "in-group pre-distribution". While the pre-distribution scheme establishing a single share key between two groups is called "cross-group pre-distribution". The in-group pre-distribution scheme could then be any of the existing pre-distribution schemes. In [19], $\{D_i\}_{i=1,2,\ldots,n}$ are defined as instances of an existing in-group pre-distribution $\{G_i\}_{i=1,2,\ldots,n}$. While $\{D'_i\}_{i=1,2,\ldots,m}$ are defined as instances of a cross-group pre-distribution $\{G'_i\}_{i=1,2,\ldots,m}$. This scheme must meet both of the following conditions: (i) each G'_i contains exactly one node of a group and (ii) $\forall\, i,j \; such\; that\; i \neq j, then\; G'_i \cap G'_j = \emptyset \; and \; |G'_i \cap G'_j| = 1$. Thus, each cross-group constitutes a potential link for any pair of two in-group.

12.2.1.2. Schemes Based on the Base Station Participation or on a Master Key Pre-distribution

Perrig et al. [23] have proposed the SPINS protocol which is based on two protocols: Sensor Network Encryption Protocol (SNEP) and µTESLA. SNEP ensures confidentiality and authentication of data exchanged between two nodes by adding 8 bytes to each message. As for µTESLA, which is an extended version of TESLA [24], it only ensures the authentication of broadcast packets. SNEP relies on the use of the RC5 encryption algorithm by using it in CTR (CounTeR) mode. At the beginning of the deployment, the network topology only allows communications between the sensor nodes and the base station. After network deployment, SPINS introduces a method for extending established trust between nodes and the base station to other communication links between regular nodes.

The protocol consists in storing a master pre-shared secret key (***X***) in the sensor node and in its base station. After network deployment, a node ***A*** and a base station ***BS*** each derive two secret keys from the master pre-shared key ($X_{A,BS} = X_{BS,A}$): the node ***A*** gets $K_{A,BS}$ and $K'_{A,BS}$ and the node ***BS*** gets $K_{BS,A}$ and $K'_{BS,A}$. The first key is used for encryption and the second is dedicated to the calculation of the message authentication code (MAC). Key derivation is done by using a pseudo-random function. Since the two nodes share the same master key and the same derivation function, each node will forward the secret keys of the other to be able to decrypt and check the received messages. Instead of sending counters with messages, ***A*** and ***B*** share two counters (one for each direction of communication). Counters are synchronized using a counter exchange protocol as shown in Fig. 12.3. C_A and C_B are the respective counters of the nodes ***A*** and ***B***. Both counters are not secret, so ***A*** can send his counter to ***B*** in a clear text. At the reception, ***B*** sends its counter accompanied by the message authentication code (MAC) of the combination of the two counters. Thus, ***A*** will be able to check the integrity of C_B. It then sends the message authentication code of the combination of the two counters to ***B*** so that it can check the integrity of C_A.

This counter sharing method saves energy in a good way by replacing the sending of counters in each message with a MAC calculation and counter synchronization. Note that this method can only be beneficial if the two communicating nodes have a lot of data to exchange. In addition, each communication needs two synchronized counters (one per communication direction), and this could overload the calculation and memory storage of the nodes.

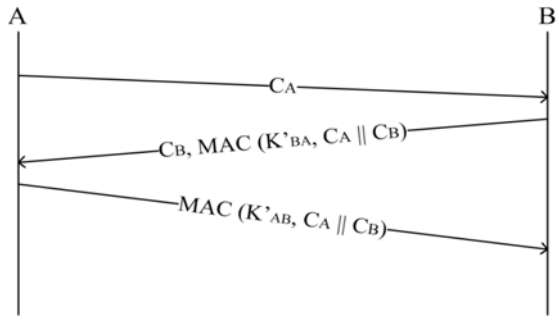


Fig. 12.3. Counter exchange protocol.

Zhu et al. [25] propose the LEAP protocol, a WSN authentication and key management protocol. LEAP relies on the use of four types of keys per node: (i) an individual key shared with the base station, (ii) a unique key shared with another node, (iii) a shared cluster key with its neighboring nodes of the same cluster, (iv) and a group shared key, with all the nodes of the network. The network controller generates an initial master key and stores it in each node memory before network deployment. After the network deployment, each node $\boldsymbol{u}$ of the network derives from this key its own master key $\boldsymbol{K_u = f_{Ki}(u)}$. Then it broadcasts a HELLO message containing its identifier to its neighbors ($\boldsymbol{u \rightarrow *: u}$). When it receives an ACK from a neighbor v $\boldsymbol{(v \rightarrow u: v, MAC (K_v, u \mid v))}$, it will be able to verify its identity by firstly computing $\boldsymbol{K_v = f_{Ki}(v)}$ and then the MAC. Once the identity of $\boldsymbol{v}$ is verified, $\boldsymbol{u}$ calculates its unique key shared with it $\boldsymbol{K_{uv} = f_{Kv}(u)}$. The node $\boldsymbol{v}$ can then calculate this key in the same way. Note that $\boldsymbol{u}$ does not need to authenticate to $\boldsymbol{v}$ because authentication will be implicit when sending encrypted messages using the common key. After a certain time defined at the beginning of the network deployment, the node $\boldsymbol{u}$ will remove the initial master key and the keys formerly calculated from its neighboring nodes (the $\boldsymbol{K_v}$). It keeps only its individual key and the unique keys shared with its neighbors. Suppose a node $\boldsymbol{u}$ is a cluster head sending a cluster key to its neighbor nodes $\boldsymbol{v_1, v_2, \ldots, v_m}$. It first generates a random key $\boldsymbol{K_u^c}$. It then encrypts it by using the key shared with each $\boldsymbol{v_i (1 \leq i \leq m)}$ before sending it in a unicast way $\boldsymbol{v_i (u \rightarrow v_i: (K_u^c)_{Ku,vi})}$. In order to distribute group keys shared by the entire network, LEAP adapts the μTESLA protocol. Finally, LEAP uses the RC5 algorithm for encryption and uses the CBC-MAC mode for message authentication.

12.2.2. Asymmetric Key Cryptographic Systems

12.2.2.1. PKI Based Schemes

The SKKE algorithm is a symmetrical protocol proposed by the ZigBee Alliance which relies on the use of a master key to create a common key (LK) shared between any two nodes of the network. Therefore, if the master key is compromised, the LK key may be compromised as well. In order to solve this issue, a new protocol has emerged: the PKKE [26] protocol based on the use of public keys infrastructure (PKI). As the creation of public keys relies on static data that has a short lifetime, and not on the confidentiality of a master key, so its compromising will not necessarily affect the LK key.

The Fig. 12.4 shows the LK key establishment following the PKKE protocol steps. This is done in four stages. The first step is to exchange static data (S_u, S_v) coupled with data generated by ***u*** and ***v*** during a short time (E_u, E_v). (S_u, S_v) is a combination of the node addresses (node identifiers) with their static public key. E_u and E_v are public key pairs randomly generated for a short time. The second step of the PKKE protocol is to use a function (KeyBitGen function) that produces the same keys on both sides. This function is based on the use of the ECMQV (Elliptic Curve Menezes-Qu-Vanstone) key sharing method specified in [27]. The third step is to derive the two keys Mackey and keyData, and to compute the MAC on both sides. The last step of the protocol consists in exchanging the computed MAC to have them checked by both ***u*** and ***v***. Once the MAC are verified, then KeyData will be the shared symmetric key.

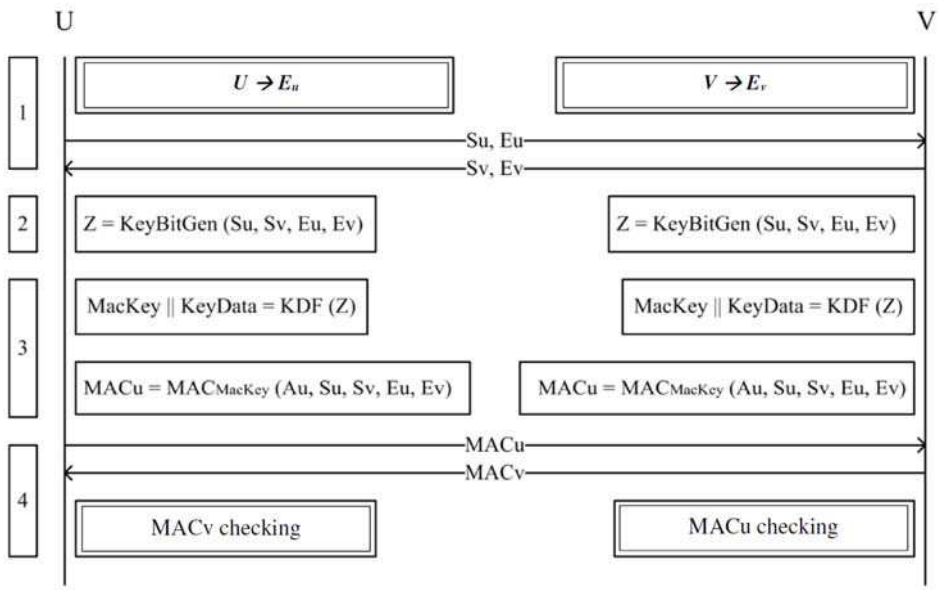


Fig. 12.4. LK key establishment following the PKKE protocol steps.

As you can see, the difference between the PKKE asymmetric protocol and the SKKE symmetric protocol lies in the use of an ECMQV asymmetric mechanism by PKKE in order to avoid the use of a pre-distributed master key which could be compromised and thus destroy the overall security of the network.

On its side, the CBKE [26] (Certificate-Based Key-Establishment) protocol aims to implement the use of digital certificates signed by a trusted certificate authority (CA). Each node of the network has a private key and a public key embedded with a certificate signed by the CA. Certificates contain specific information such as network ID, validity date, node class, and so on. The CBKE protocol has a structure identical to that of PKKE but here certificates ($CERT_u$, $CERT_v$) are used instead of the public keys ($\boldsymbol{E_u}$, $\boldsymbol{E_v}$). The signature of the CA is verified upon reception of the certificates by $\boldsymbol{u}$ and $\boldsymbol{v}$.

12.2.2.2. Node Identity Based Schemes

The goal of the author of the IBC (Identity-Based Cryptography) method [27] was to introduce a new type of cryptographic methods which are able to provide the same cryptographic services as a PKI without the need: (i) to exchange private/public keys, (ii) to store created key directories in node memories, (iii) nor to use the services of a trusted third party. The IBC protocol is based solely on the information of the identity of the person (name combination, social security number, telephone number, etc.). This method exploits the principle of issuing smart cards containing a private key that corresponds to the personal information of the user itself. So, if Bob wants to send a message to Alice, then he has only to sign the message with his private key and then encrypts the result with the public information of Alice before sending it. Upon the reception of the message, Alice decrypts it with her private key and checks Bob's signature using Bob's personal public information. The system implementation requires a public key schema to generate the private keys. This system has been used to develop new encryption systems such as the Identity-Based Encryption (IBE) cryptography system [27].

The PKKE and CBKE protocols use the identity of the nodes in their key establishment methods. The goal is to use these identities to create a unique shared key between each pair of nodes in the network. However, the creation of the shared key is realized with interactions between the two nodes. That is to say that these methods require sending and receiving multiple messages from both sides before creating the key. Which results in wasting too much energy. In order to save the energy of the nodes, several Identity-Based Non-Interactive Key Distribution Schemes (ID-NIKDS [28]) have been proposed in order decrease or to suppress these interactions.

Oliveira et al. [28] have proposed a ID-NIKDS method which uses PBC coupled with the IBC system to generate keys based on the identity of the WSN nodes. The main idea is that each network node has a unique identifier and a private key, so that any node would be able to derive a unique common secret key with another node by only knowing its identifier.

12.2.3. Comparative Analysis

In the previous parts, we presented several cryptographic schemes. In this section we will now give a brief comparative study of these protocols (Table 12.1). Here, several challenges or criteria (Fig. 12.5) are taken into account in order to compare different methods of key management.

Table 12.1. WSN key cryptographic management schemes comparison.

<table>
<tr><th colspan="3" rowspan="3"></th><th rowspan="3">Scaling</th><th rowspan="3">Connectivity</th><th colspan="4">Resistance against attack</th><th colspan="3">Cost of resource used</th></tr>
<tr><th rowspan="2">Eavesdropping attack</th><th rowspan="2">Communications disruption</th><th rowspan="2">Resources exhaustion</th><th rowspan="2">Node capture</th><th rowspan="2">Storing capacity memory</th><th rowspan="2">Computing and energy consumption</th><th rowspan="2">Key renewal and revocation processes</th></tr>
<tr></tr>
<tr><td rowspan="6">Key management schemes</td><td rowspan="4">Symmetrical</td><td>Chan et al.</td><td>L</td><td>M</td><td rowspan="4" colspan="4">L</td><td>H</td><td>L</td><td rowspan="4">H</td></tr>
<tr><td>Chan and Perrig (PIKE)</td><td>No</td><td>H</td><td>M</td><td>H</td></tr>
<tr><td>Perrig et al. (SPINS)</td><td>L</td><td>M</td><td rowspan="2">L</td><td rowspan="2">L</td></tr>
<tr><td>Zhu et al. (LEAP)</td><td>L</td><td>M</td></tr>
<tr><td rowspan="2">Asymmetrical</td><td>ZigBee (PKKE)</td><td>M</td><td>L</td><td rowspan="2" colspan="3">H</td><td rowspan="2">M</td><td>H</td><td>M</td></tr>
<tr><td>Oliveira et al. (TinyPBC)</td><td>M</td><td>H</td><td>M</td><td>L</td></tr>
<tr><td colspan="3"></td><td colspan="8">Legend: High (H), Medium (M), Low (L)</td></tr>
</table>

The proposed key management methods must take into account the fact that the nodes have been deployed to collect the information. So, the nodes need their memory space to store collected data, and also need their embedded energy to ensure long application lifetime as much as possible. Moreover, cryptographic solutions must also be flexible, dynamic and suitable for scaling (scalability). Managing new nodes insertion in the network and physical node losses during network operation must be done in a manner that maintains the overall security level. Therefore, any cryptographic scheme must also offer a very good resistance against attacks. In case of node capture, for example, the adversary can use the stored information to set up other attacks against the network. The key management method used must be able to detect compromised nodes and authenticate network nodes before distributing keys. Another important challenge to take into account is keys renewal and revocation management processes. An expired key or a key discovered by an opponent must be revoked in timely manner. The keys of the secure

links must also be renewed periodically. The key distribution method must be able to ensure and maintain a good quality of network connectivity. Because, in the case of the loss of a node, the communications between of the remaining nodes could be negatively affected. Thus the key management methods must take into account this issue:

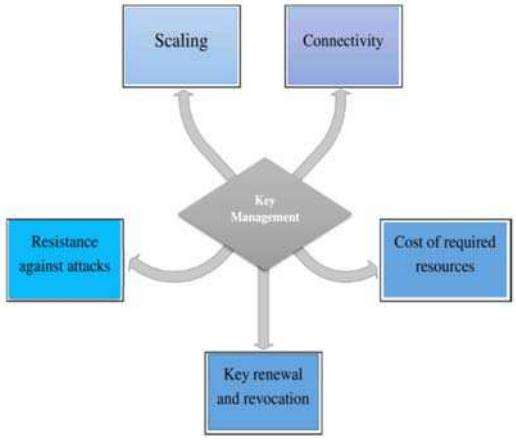


Fig. 12.5. Key management protocol challenges.

- PIKE, which is a probabilistic scheme, consumes little energy and does not require lot of computing resources. However, the large size of key sets stored in the nodes memory before deployment make this scheme among the most expensive symmetric schemes in terms of memory occupancy. On the other hand, it can not resist some attacks such as physical node capture attack. Finally, while PIKE method provides better connectivity between the nodes of the network, it is difficult to scale with a high energy consumption due to the number of messages exchanged for key establishment.

- SPINS and LEAP schemes use master keys in its key establishment process. This reduces the required capacity for keys storage in the memory of the nodes. However, resistance to attacks is low. Since the master key may be compromised at any time, provided keys after deployment using this master key may be compromised as well. By adopting a symmetrical system, they require less computing resources. We can notice that, these symmetric schemes are expensive in the operations (if they exist) of keys renewal and revocation since they use secret keys to exchange other secret keys. The problem is simpler in asymmetric schemes since public keys do not need to be secret.

- As we can see, the TinyPBC scheme is a suitable asymmetric cryptographic scheme. It resists most known attacks in WSN. Using coupling to establish a unique shared key between two nodes helps in reducing the need for large storage capacity in node memory. In addition, the creation of this key is performed without any interaction between the nodes, which contributes to save processing time and energy consumption.

Otherwise, PKI and certificates based schemes remain more expensive in terms of processing time and energy consumption.

– Finally, the comparison between symmetrical and asymmetrical schemes may differ depending on the desired level of network security. Symmetric schemes can be chosen for their computational speed, while we may prefer the asymmetrical ones for their good resistance against attacks.

We have presented, in this section, some backgrounds and related works in the field of WSN cryptographic keys management mechanisms. The lack of infrastructure such as PKI in the WSN has forced nodes to not trust the network and to create secure paths from the data source to the base station. Works like [28] have used node identities and the coupling principle to reduce or eliminate interactions between nodes for key establishment in order to be resistant against as many attacks as possible. However, until today there are not really complete and dynamic solutions that would be easily adaptable to WSN. The methods we have studied in this section have great advantages when they are dedicated to particular network applications and topologies. However, it is difficult, if not impossible, to generalize them as such to any type of application or to make them truly versatile and universal. In what follows, we will present our contribution, which is proposing a new mechanism for cryptographic key management aimed at securing end-to-end data transmission in WSN.

12.3. LESDTP: A New Key Management Protocol

According to the previous background study, the main goal of Low Energy Secure Data Transmission Protocol (LESDTP) is to provide a key management scheme in order to add more security features for overall communications in wireless sensor networks, including network operations such as clustering process.

12.3.1. Protocol Overview

LESDTP uses the mechanisms of symmetric key management systems [29] with a deterministic method of pre-distribution of random keys to ensure the confidentiality of data, and the message authentication code (MAC) cryptosystem [29] to ensure the authenticity of the data source node and the integrity of the received information. Then, firstly, it would be possible to generate a key for each node of the network. Hence, a key range resulting from a finite set ***{P}*** generated by the Base Station (BS) will be allocated. This allocation would be done in such a way that any two subsets of ***{P}*** would have a high probability of having a common key ***"k"***. As a result, any two nodes sharing a common key would be able to exchange secure messages. This common key is used to compute the Message Authentication Code (MAC) [29] and data encryption for any communication between the two nodes.

12.3.2. Assumptions

LESDTP is based on the following assumptions:

- The autoconfigurable network is static. So, there is no mobile node in the network;
- The processing, communication, power and storage capacity of the Base Station (BS) is greater than that of other nodes in the network;
- The network is homogeneous: all nodes, except the BS, are similar in their processing, communication, energy and storage capacity;
- Nodes are using a common clock, and are synchronized;
- The deployment topology is done in a randomly manner [5, 6]: the neighbors of any node are not known before deployment;
- The communication channels are bidirectional: if a node A can receive a message from a node B, then B can send a message to A;
- The routing protocol used is L2RP [5, 30, 31];
- Each node is pre-loaded with a unique ID, a keychain and a secret key ***"k_s"*** before deployment;
- The pre-distribution of keys is deterministic;
- Cryptography is based on symmetric encryption: The same key is used for a pair of nodes to encrypt and decrypt messages exchanged between them using a symmetric encryption algorithm.

12.3.3. Notations

The following symbols are used to describe the LESDTP process (Table 12.2).

Table 12.2. LESDTP protocol symbols.

Symbols	Meaning	Symbols	Meaning
BS	Base Station	NC	Total number of simple nodes in a cluster
X	Total number of network nodes	N_i	Simple node number i
X_j	Node number j	$P(N_i)$	Keychain P of the simple node number i
ID	ID of a node	D	Data to be transmitted
{P}	Finished set of keys	MAC()	MAC calculated by a node
$P(X_j)$	Keychain P of the node number X_j	TT	Data transmission time
K	Key shared by all nodes in the network	PP	Physical phenomenon
k_c	Common key between two nodes	F()	Encryption function
C	Total number of clusters	D()	Decryption function
C_a	Cluster number a	$\rightarrow$	Message transmission
CH	Cluster Head	$\|\|$	Concatenation operator

12.3.4. Phases of the LESDTP Protocol

The LESDTP process runs in several phases: key management and network implementation phase, communication phase and key maintenance phase.

12.3.4.1. Key Management and Network Implementation Phase

This initial phase consists of keys pre-distribution (Algorithm 1), random network nodes deployment, clustering operations, shared key discovery, and secure link establishment.

12.3.4.1.1. Keys Pre-distribution

Before the network deployment, each network node X_i is pre-loaded with an ***ID***, a secret key ***k***, and a key ring ***p*** selected from a large set ***P*** of keys by the BS. This one then stores on a table all the ***IDs*** of the nodes. The number of keys of the set ***P*** is chosen so that two random subsets of ***P*** would have a probability of sharing a same key ***"k_c"***.

Algorithm 1. Keys pre-distribution.

{P} finite set of keys;

X the total number of nodes in the network;

k the secret key shared by all nodes;

ID the identifier of a node;

P(X_j) the keychain of node j;

Begin

For $j \rightarrow 1$ to X

BS generates ***{P}***

BS generates ***k***

Xj ← BS: ***ID(X_j)***

X_j ← BS: $P(X_j), P(X_j) \in \{P\} / P(X_1) \cap P(X_2) \cap \ldots \cap P(X_X) = k_c$

X_j ← SB: ***k***

BS← $\sum ID(X_j)$

Random deployment of network nodes;

End

12.3.4.1.2. Clustering Process

After a random deployment of the nodes, then the network runs a 1-hop clustering process [5, 32] so that each cluster member of a cluster is a 1-hop neighbor of its clusterhead (CH) (Fig. 12.6). When clusters are formed, each simple node sends its data to its respective clusterhead. Upon data reception from all cluster members, then the CH performs data processing functions (aggregation and compression, etc.), and then transmits data towards the base station. A node decides which cluster to join based on the power of the received signals, by using the LQI-DCPSec clustering algorithm proposed in [33].

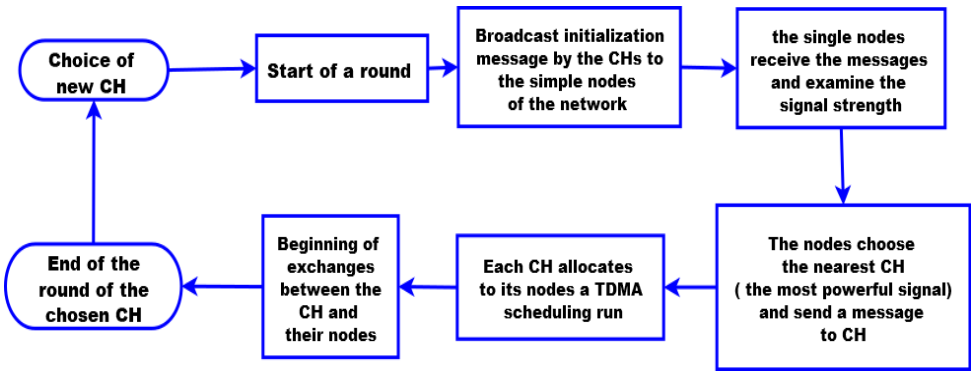


Fig. 12.6. Clustering process flowchart.

After the clustering process, each node which is designated CH then broadcasts an initialization message to its cluster members. The initialization message contains the ID of the CH, encrypted with the secret key $\boldsymbol{"k_s"}$. Since, the clustering process is based on the link quality indicator (LQI) [5, 33], then each node would need less energy to transmit data to its corresponding CH. Thus energy efficiency is guaranteed.

Then, each cluster member transmits a message containing its node ID, the CH ID, and a header. After receiving messages from all cluster members, then the CH allocates to each node of its cluster a slot time based on a TDMA scheme (Time Division Multiple Access). And it is only during the allocated slot time that the nodes could be able to transmit data to the CH.

12.3.4.1.3. Shared Key Discovery and Secure Link Establishment

Once the network is set up and the clusters are formed, each cluster member sends its key string and ID to its CH. Then, the CH retrieves the shared key, $\boldsymbol{"k_c"}$, and sends it back to the corresponding node. Such communications are done in a secure way because exchanges are encrypted using the secret key $\boldsymbol{"k"}$. This shared key is also used as the session key of the link between the two nodes. CHs proceed in the same way with the BS.

After the shared key discovery phase, secure links are then established between each cluster member and its clusterhead (CH), and also between each CH and the BS. Since,

nodes are randomly deployed, CHs do not necessary form a backbone for routing process (Algorithm 2). So we use L2RP [5, 30, 31] as routing protocol, which could lead to use both CHs and non clusterhead nodes for data transmission between CHs and the BS [5, 31-34]. The network then becomes a connected graph formed of several secure links.

Algorithm 2. Shared key distribution.

Begin

For a → 1 to C

M_1 = [***ID*** [CH (C_a)] ‖ p [CH (C_a)]];

CH (C_a) → BS: F (M_1); # the CHs send their keychain to the BS

BS → CH (C_a): F (k_c); # BS returns to each CH the shared key

CH (C_a): D (k_c); CH (C_a) ← k_c;

End for

For a → 1 to C

For i → 1 to N_C

M_2 = [ID (N_i) ‖ p (N_i)];

N_i → CH (C_a): F (M_2); # nodes sending key to CH

CH (C_a) → N_i (C_a): F (k_c);

CHs return to each node of its cluster the shared key

N_i (C_a): D (k_c);

N_i (C_a) ← k_c

End for

End for

#The network then becomes a connected graph with secure links

End

12.3.4.1.4. Data Transmission Phase

There are two different types of communication: one between a cluster member node and its CH; and another one between the CHs and the base station (BS). Each CH hold a table containing all the identifiers (IDs) of each member of its cluster. Similarly, the BS also holds a table with the IDs of all the CHs of the entire network.

– **Communication between a cluster member and its CH:** The simple nodes of a cluster collect the physical phenomena and then send data directly and periodically to its CH. In order to guarantee the authentication of the source node and to secure communication, data transmission phase will involve the date and time of transmission, the ID of the collector node and the MAC generated with the shared

secret key with the CH (Algorithm 3). Collected data are encrypted before transmission. Upon reception, the CH will firstly check the timestamp to ensure data freshness by discarding replayed data before decrypting the received data. If the timestamp is correct, it then calculates the MAC with the same key and compares it with the received one. If they are identical, the data will be accepted. The source node is then authentic and the data integrity is also correct. Otherwise data would be rejected and the source node would be considered as corrupted (Fig. 12.7, 12.8).

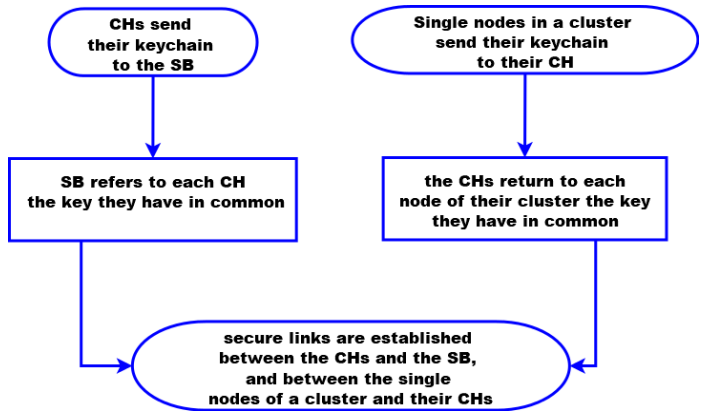


Fig. 12.7. Flowchart of key discovery and secure link establishment.

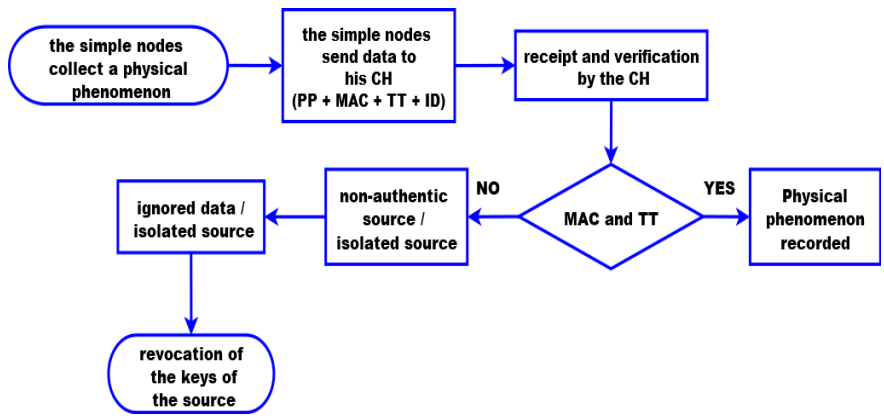


Fig. 12.8. Flow chart of the communication between a simple node and its CH.

Algorithm 3. Encryption between regular node and CH.

PP = physical phenomenon

C_a = cluster number a

N_{iCa} = simple node i of the cluster a

C = total number of cluster

N_{Ca} = total number of nodes in the cluster Ca

D = data to send

Begin

For a → 1 to C

For i → 1 to N_{Ca}

$D(N_{iCa}) = [PP(N_{iCa}) \| TT(N_{iCa}) \| MAC(k) \| ID(N_{iCa})]$

$CH(C_a) \leftarrow F[D(N_{iCa})]$

End for

End for

End

– **Data transmission between a CH and the BS:** after reception and validation of the received data from all its cluster members, each CH should compute some other operations like data aggregation [2]. Then, it calculates a new MAC with the shared secret key with the base station. After that, it concatenates aggregated data (resulting from collected physical phenomenon) with the MAC, the date and time of transmission and the ID. Finally, CH performs encryption and then sends resulting data to the BS (Algorithm 4). Upon reception and before using received data, the BS performs some control operations like: decryption, time stamp, MAC and source validation.

Algorithm 4. Encryption between CHs and the BS.

PP = physical phenomenon

C_a = cluster number a

N_{iCa} = simple node i of the cluster a

C = total number of cluster

N_{Ca} = total number of nodes in the cluster C_a

D = data to send

Begin

For a → 1 to C;

$CH(C_a)$: $D(D(N_{iCa}))$;

$CH(C_a) \leftarrow PP(N_{iCa})$;

$D[CH(C_a)] = [PP(N_{iCa}) \| TT[CH(C_a)] \| MAC(k) \| ID[CH(C_a)]]$;

$BS \leftarrow F(D[CH(C_a)])$;

End for;

End.

12.3.4.2. Key Maintenance Phase

The following operations could be performed during this phase: keys renewal, keys revocation, new node insertions.

12.3.4.2.1. Keys Renewal

In order to maintain a high security level of our protocol, the ***{P}*** key set will be renewed by the BS after a fixed period. This will also lead to the renewal of the key string of all the nodes of the entire network. After the renewal of ***{P}***, the BS will send to each node its new keychain. The recovery of the shared key discovery and secure link establishment phase will be necessary at this step.

12.3.4.2.2. Keys Revocation

The keys revocation phase is triggered each time a network node is compromised. If the compromised node is a member of a cluster, its CH removes its keys string from memory and quarantines it so that it would not be eligible as CH in next rounds.

On the other hand, if the compromised node is a CH, the base station deletes its keys string, triggers a cluster head election process between the non-compromised cluster members. Cluster members are notified that CH is compromised, then they ignore the messages from the corrupted CH and quarantine it. So, each cluster member sends its ID and keychain to the new CH to discover the shared secret key and then begin to establish secure links.

12.3.4.2.3. New Node Insertions

Before insertion in the network, a new node is pre-loaded with an ID and the shared secret key ***"k_s"***. After its deployment on the network, the base station informs all CHs of the arrival of the new node. CHs send an initialization message to the new node. The new node then chooses which CH to join based on the power of the received signal as explained in the clustering phase. Finally, it sends back to its CH its ID and its keychain to establish the discovery phase of the shared secret key and secure link establishment.

12.4. Evaluation of LESDTP Protocol

In this section, we will evaluate the performance of our LESDTP protocol. But before that, one can easily understand, upon the above description, that this protocol could be used in many applications of various fields. Nevertheless, LESDTP does not support sport applications in which mobile athletes are equipped with sensor nodes. In such applications, nodes are not static. The following Table 12.3 summarizes the LESDTP application areas of autoconfigurable networks like wireless sensor networks or internet of things.

Table 12.3. Application fields of LESDTP protocol.

Application domain	Applicability
Agricultural	Yes
Medicine	Yes
Meteorology	Yes
Military	Yes
Environment	Yes
Sport	No
Transport	Yes

12.4.1. Security Services Offered by the LESDTP Protocol

Now let's look at the security features offered by this protocol. As a result, we will then evaluate its resistance to some attacks. Finally, we will make a comparison with some protocols of the state of the art.

Our LESDTP protocol provides the following security features: data integrity, source node authentication, data confidentiality, data freshness. It also provides resistance to some attacks like that listening, sniffing, sinkhole and sybil attacks [1, 2].

12.4.1.1. Data Integrity and Source Node Authentication

The computed Message Authentication Code (MAC), combined with the encryption of data sent by the issuer, helps in ensuring the data integrity and source authentication. When an attacker attempts to corrupt data, the cluster heads or the base station will detect it, since the MAC is calculated upon data reception. If the MACs are identical, then data integrity is correct and the source is genuine. Otherwise, the sending node will be considered as compromised.

12.4.1.2. Data Confidentiality

All communications inside any cluster and also between any CH and the BS are encrypted. This approach ensures the data confidentiality. Moreover, keys renewal and keys revocation processes in the event of a compromised node will help in maintaining its high level of security.

12.4.1.3. Data Freshness

In order to ensure data freshness security feature, clock nodes are synchronized and then sent messages are stamped with the sender’s date and time. A time counter is also added to the packet, and keys are regenerated periodically in order to avoid replay attacks.

12.4.2. LESDTP's Resistance to Attacks

Given its inherent properties, LESDTP protocol also gives good resistance to the following attacks:

- **Listening or sniffing attacks** [1, 2]: In autoconfigurable wireless networks, an attacker could usually listen to communications and eventually intercept packets exchanged between a regular node and its cluster head (CH) or between a CH and the base station (BS). However, in our case, it would not be possible to recover the contents because of the encryption of all data during the communications. Here again, the fact that the keys are regenerated periodically helps in maintaining a good level of security by countering cryptanalysis attacks like ciphertext-only attack, known-plaintext attack, chosen-plaintext attack, or chosen-ciphertext attack.

- **Sinkhole attack** [1, 2]: Using symmetric keys and a MAC assigned to each data in order to fully authenticate the transmitters offers to LESDTP the protection feature related to this kind of attack. Moreover, the clustering process and communications inside each cluster are encrypted with keys, which are periodically changed. So, it would be very difficult for a malicious node to achieve a sinkhole attack in this network.

- **Sybil attack** [1, 2]: In Sybil attack, the attacker bypasses the reputation system by creating a large amount of malicious identities and using them to have a disproportionate influence. Node identification and the use of unique shared secret key for peer to peer communications are some of LESDTP properties allowing to establish secure communication links in order to fight against this kind of attack.

12.4.3. LESDTP Performance Analysis

12.4.3.1. Attacker Model

We consider an attacker who is in the neighborhood of clusters and whose purpose is to intercept the messages in order to retrieve required information to enter the network. This attacker would also like to modify exchanged data in the network and to attempt adding malicious nodes in order to disrupt communications and network proper functioning.

12.4.3.2. LESDTP Analysis Relatively to This Attacker Model

Since all messages sent over the network are coupled with a MAC, if the attacker manages to intercept a message and tries to tamper with or delete it, the CH will be able to detect this by recalculating the MAC and comparing it with the one received with the message. Thus our attacker will not succeed in altering data.

If the attacker compromises a regular node, he will not be able to communicate with the CH. A secure link will not be established between the malicious node and the CH because it ignores the required shared secret key for the communication.

If the attacker captures a node, he will not be able to extract the data stored in the node's memory. One of the LESDTP properties is the storage encryption which would help in preventing the attacker from recovering the cryptographic information (key, encryption algorithm, etc.).

If an attacker compromises a CH, it is not sure that the comprised CH will still remain cluster head at another date because LESDTP uses a dynamic cluster formation protocol in which the CHs are selected in a rotating manner.

As we can see, LESDTP resists perfectly to the defined attacker model since it ensures confidentiality and data integrity as well as source node authentication.

12.4.4. LESDTP Comparison with Some Protocols of the State of the Art

In this section, LESDTP is compared with some deterministic random key pre-distribution schemes.

PIKE [34] (Peer Intermediaries for Key Establishment) is a protocol which minimizes key storage in node memory before deployment. PIKE proposes using the regular network nodes as trusted intermediaries instead of the base station. This is proposed in order not to exhaust the node resources near the base station. However, this solution could have a disadvantage, because if the trusted intermediate nodes between A and B are captured, then A will neither be able to share a secret key with B, nor to establish a secure link with B.

Liu et al. [35] have proposed a similar approach to *Chan and Perrig* [34] because it also uses logic grids. However, their work is based on the known location of deployed nodes to form clusters.

The comparison is summarized in the Table 12.4. We can notice that LESDTP offers the most effective resistance to attacks. All nodes in the network are authenticated and data integrity is ensured. It is also more energy efficient and more scalable.

12.5. Conclusion and Future Works

Security needs related to autoconfigurable networks such as wireless sensor networks differ from one application domain to another and still remain to this day, a true challenge both in theoretical and practical aspects. Without any security feature, the use of autoconfigurable wireless networks in any field of application would have undesirable consequences. Sharing secure data involves establishing and distributing keys to encrypt messages and authenticate sources nodes. We have studied several key management

protocols. We have been particularly interested in deterministic protocols based on symmetric cryptography and with keys pre-distribution. Then, we have proposed a new protocol (LESDTP) that ensures and maintains communications security while minimizing energy consumption over the network and provides security features.

Table 12.4. Comparative table of LESDTP with some deterministic pre-distribution key schemes.

		Comparison criteria							
		Scaling	**Connectivity**	**Resistance against attacks**			**Cost of resources used**		
				Eavesdropping attack	**Resources exhaustion**	**Node capture**	**Memory (key storage)**	**Computing and energy consumption**	**Key renewal and revocation**
Schemes	PIKE [34]	no	H	L			M	H	H
	Liu and al. [35]	no	H	M			M	H	H
	LESDTP	yes	H	H			M	L	L
Legend: High (H), Medium (M), Low (L)									

In addition, since node mobility is an important property for some autoconfigurable wireless network applications, we will be interested in future work to produce a revised version of LESDTP that takes into account this feature. We will also propose simulation scenarios to implement a comparative study of the revised version of LESDTP supporting node mobility with some studied protocols. This will validate and confirm the results outlined in this chapter.

References

[1]. C. Diallo, A. Sawaré, M. T. Sow, Security issues and solutions in wireless sensor networks, *International Journal of Computer Science and Information Security*, Vol. 15, Issue 3, March 2017, pp. 6-17.

[2]. C. Diallo, Security issues and solutions related to data aggregation process in WSN, *International Journal of Computer Science and Network Security*, Vol. 17, Issue 4, April 2017, pp. 59-71.

[3]. H. M. Alzaid, Secure data aggregation in wireless sensor networks, PhD Thesis, *Queensland University of Technology*, 2011.

[4]. C. Kaufman, R. Perlman, M. Speciner, Network Security: Private Communication in a Public World, 2nd Ed., *Prentice Hall*, 2002.

[5]. C. Diallo, Techniques d'amélioration du routage et de la formation des clusters multi-sauts dans les réseaux de capteurs sans fil, *Doctorat de Télécom SudParis & Université Pierre et Marie Curie (Paris VI)*, Sept. 2010.

[6]. M. T. Sow, C. Diallo, Energy over-consumption induced by securing network operations, in *Proceedings of the 2nd International Conference on Frontiers of Sensors Technologies (ICFST'17)*, Shenzhen, China, April 2017, pp. 154-160.

[7]. M. Turkanovic, B. Brumen, M. Hölbl, A novel user authentication and key agreement scheme for heterogeneous ad-hoc wireless sensor networks, based on the Internet of Things notion, *Ad Hoc Networks*, Vol. 20, 2014, pp. 96-112.
[8]. C. Manifavas, G. Hatzivasilis, K. Fysarakis, K. Ranto, Lightweight cryptography for embedded systems – A comparative analysis, *LNCS*, Vol. 8247, 2014, pp. 333-349.
[9]. R. Amin, G. P. Biswas, A secure light weight scheme for user authentication and key agreement in multi-gateway based wireless sensor networks, *Ad Hoc Netw.*, Vol. 36, Issue P1, January 2016, pp. 58-80.
[10]. Y. Lu, L. Li, H. Peng, Y. Yang, An energy efficient mutual authentication and key agreement scheme preserving anonymity for wireless sensor networks, *Sensors*, Vol. 16, Issue 6, 2016, 837.
[11]. M. A. Salam, Detectability of man-in-the-middle attacker in mobile sensor networks, *International Journal of Advanced Computer Science*, Vol. 3, Issue 2, 2013, pp. 99-103.
[12]. S. Glass, V. Muthukkumurasamy, M. Portmann, Detecting man-inthe-middle and wormhole attacks in wireless mesh networks, in *Proceedings of the International Conference on Advanced Information Networking and Applications (AINA'09)*, 2009, pp. 530-538.
[13]. K. Venkatraman, J. Daniel, G. Murugaboopathi, Various attacks in wireless sensor network: Survey, *International Journal of Soft Computing and Engineering (IJSCE)*, Vol. 3, 2013, pp. 208-211.
[14]. N. Alajmi, Wireless sensor networks attacks and solutions, *International Journal of Computer Science and Information Security*, Vol. 12, Issue 7, July 2014.
[15]. N. A. Niang, C. Diallo, Low energy secure data transmission protocol for autoconfigurable networks, in *Proceedings of the IEEE International Conference on New Trends in Engineering & Technology (ICNTET'18)*, September 2018.
[16]. L. Eschenauer, V. D. Gligor, A key-management scheme for distributed sensor networks, in *Proceedings of the 9th ACM Conference on Computer and Communications Security (CCS'02)*, New York, NY, USA, 2002, pp. 41-47.
[17]. H. Chan, A. Perrig, D. Song, Random key predistribution schemes for sensor networks, in *Proceedings of the Symposium on Security and Privacy (S&P'03),* 2003, pp. 197-213.
[18]. H. Chan, A. Perrig, PIKE: In sensor networks, in *Proceedings of the IEEE 24th Annual Joint Conference of the Computer and Communications Societies (INFOCOM'05)*, Vol. 1, 2005, pp. 524-535.
[19]. D. Liu, P. Ning, W. Du, Group-based key predistribution for wireless sensor networks, *ACM Trans. Sen. Netw*., Vol. 4, Issue 2, 2008, pp. 11:1-11:30.
[20]. S. A. Çamtepe, B. Yener, Combinatorial design of key distribution mechanisms for wireless sensor networks, in *Proceedings of the European Symposium on Research in Computer Security (ESORICS'04)*, 2004, p. 293-308.
[21]. J. Lee, D. R. Stinson, A combinatorial approach to key predistribution for distributed sensor networks, in *Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC'05)*, Vol. 2, 2005, pp. 1200-1205.
[22]. S. Ruj, B. Roy, Key predistribution schemes using codes in wireless sensor networks, in *Proceedings of the International Conference on Information Security and Cryptology (Inscrypt'08)*, 2009, pp. 275-288.
[23]. A. Perrig, R. Szewczyk, J. D. Tygar, V. Wen, D. E. Culler, SPINS: Security protocols for sensor networks, *Journal of Wireless Networks*, Sept. 2002, Vol. 8, Issue 5, pp. 521-534.
[24]. A. Perrig, R. Canetti, J. Tygar, D. Song, Efficient authentication and signing of multicast streams over lossy channels, in *Proceedings of the IEEE Symposium on Security and Privacy (S&P'2000)*, 2000, pp. 56-73.
[25]. S. Zhu, S. Setia, S. Jajodia, LEAP+: Efficient security mechanisms for large-scaledistributed sensor networks, *ACM Trans. Sen. Netw*., Vol. 2, Issue 4, 2006, pp. 500-528.

[26]. ZigBee, ZigBee Smart Energy Specification, 075356r15, *ZigBee Standards Organization*, 2008.
[27]. Standards for Efficient Cryptography, SEC 1 (working draft), Ver. 1.7, Elliptic Curve Cryptography, *Certicom Research*, 2006.
[28]. L. B. Oliveira, et al., TinyPBC: Pairings for authenticated identity-based non-interactive key distribution in sensor networks, *Computer Communications*, 34, 2011, pp. 485-493.
[29]. A. Bachir, A. Ouadjaout, L. Khelladi, M. Bagaa, N. Lasla, Y. Challal, Information security in wireless sensor networks, in Handbook/Encyclopedia on Ad Hoc and Ubiquitous Computing (P. Agrawal Dharma, X. Bin, Eds.), *World Scientific*, 2009.
[30]. C. Diallo, WSN routing improvement techniques, Chapter 1, in Advances in Networks, Security and Communications: Reviews, Vol. 1, Dec. 2017, *IFSA Publishing*, pp. 23-87.
[31]. C. Diallo, M. Marot, M. Becker, Link quality and local load balancing routing mechanisms for wireless sensor networks, in *Proceedings of the* 6th *Advanced International Conference on Telecommunications (AICT'10)*, Barcelona, Spain, 2010, pp. 306-315.
[32]. C. Diallo, Deployment strategies and clustering protocols efficiency, *Sensors & Transducers*, Vol. 213, Issue 6, June 2017, pp. 9-23.
[33]. C. Diallo, M. T. Sow, LQI-DCPSec: Secure distributed d-cluster formation in wireless sensor networks, in *Proceedings of the 10th International Wireless Internet Conference (WiCON'17)*, December 2017, pp. 210-220.
[34]. H. Chan, A. Perrig, PIKE: In sensor networks, in *Proceedings of the IEEE 24th Annual Joint Conference of the Computer and Communications Societies (INFOCOM'05)*, Vol. 1, 2005, pp. 524-535.
[35]. D. Liu, P. Ning, W. Du, Group-based key predistribution for wireless sensor networks, *ACM Trans. Sen. Netw.*, Vol. 4, Issue 2, 2008, pp. 11:1–11:30.

Physical and Chemical Sensors: Design, Applications & Networks

Sergey Y. Yurish, Editor

The 7th volume of popular *'Advances in Sensors: Reviews'* Book Series entitled *'Physical and Chemical Sensors: Design, Applications & Networks'* contains 12 chapters with sensor related state-of-the-art reviews and descriptions of latest achievements written by 34 authors-experts from academia and industry from 12 countries: Germany, Hong Kong, India, Mexico, Poland, Portugal, Qatar, Senegal, Spain, Turkey, Ukraine and USA.

The coverage includes current developments in physical sensors (MEMS flow sensors, load sensors and gyroscopes); chemical sensors for humidity in gases and anti-cancer agents; readout circuitry for various PWM output sensors and transducers; and autoconfigurable wireless sensor networks.

Similar to the all previously published volumes this book is available in two formats: electronic (pdf) with full-colour illustrations and print (paperback).

Like all volumes from this Book Series, the volume 7 has been also organized by topics of high interest. In order to offer a fast and easy reading of each topic, every chapter in this book is independent and self-contained. All chapters have the same structure: first an introduction to specific topic under study; second particular field description including sensing or/and measuring applications. Each of chapter is ending by well selected list of references with books, journals, conference proceedings and web sites.

The *'Advances in Sensors: Reviews'* ensures that our readers will stay at the cutting edge of the field and get the right and effective start point and road map for the further researches and developments. By this way, they will be able to save more time for productive research activity and eliminate routine work.

With the unique combination of information in this volume will be of value for scientists and engineers in industry and at universities, to sensors developers, distributors, and end users.

ISBN 978-84-09-14511-9



