PEOPLEs' DEMOCRATIC REPUBLIC of ALGERIA
Ministry of Higher Education and Scientific Research
Ferhat Abbas University Setif 1
Faculty of Sciences
Department of Computer Science


# Security in Internet of Things

A thesis presented by :

**Yasmine HARBI**

As a requirement to aim for the degree of

$3^{rd}$ cycle Doctorate in computer science

Approved by supervisory committee :

| | | |
|---|---|---|
| Prof. Nadjet KAMEL | Ferhat Abbas University Setif 1 | President |
| Prof. Allaoua REFOUFI | Ferhat Abbas University Setif 1 | Supervisor |
| Prof. Zibouda ALIOUAT | Ferhat Abbas University Setif 1 | Co-supervisor |
| Dr. Mohamed Amine FERRAG | 8 Mai 1945 University Guelma | Examinator |
| Dr. Abdelmalek BOUDRIES | Abderrahmane Mira University Bejaia | Examinator |

2020 / 2021

**“The way of success is the way of continuous pursuit of knowledge.”**

- Napoleon Hill -

**Abstract**

The Internet of Things (IoT) is an emerging wave of Internet that has drawn a lot of attention in recent years. It has made dramatic change of community life where things in IoT can take a wide variety of forms, from tiny to large objects. The explosive deployment of IoT is faced by security and privacy issues, which will have significant risks to go along with the potential benefits of the IoT. Securing such systems raises many challenges especially in resource-constrained, heterogeneous and large-scale environments. The main objective of this thesis is to overcome the security and privacy issues surrounding the IoT at different layers. In this context, we propose three efficient and robust security schemes for IoT systems to thwart attacks from both the physical world and cyberspace. We evaluate the performance and security of our proposed solutions using Burrows-Abadi-Needham (BAN) logic, Automated Validation of Internet Security Protocols and Applications (AVISPA) tool and Network Simulator 3 (NS-3). The obtained results show that our proposed techniques are secure, efficient, and suitable for IoT systems compared to recent related methods.
***Keywords*** IoT, Wireless Sensor Networks, Elliptic Curve Cryptography, Lightweight encryption, Blockchain, Authentication, Confidentiality, Trust.

**Résumé**

L'Internet des objets (IoT) est une vague émergente d'Internet qui a beaucoup attiré l'attention ces dernières années. Cela a radicalement changé la vie de la communauté ou les objets dans l'IoT peuvent prendre une grande variété de formes, des objets minuscules aux grands objets. Le déploiement explosif de l'IoT est confronté à des problèmes de sécurité et de confidentialité, qui présenteront des risques importants qui vont de pair avec les avantages potentiels de l'IoT. La sécurisation de tels systèmes pose de nombreux défis, en particulier dans les environnements à ressources limitées, hétérogènes et à grande échelle. L'objectif principal de cette thèse est de surmonter les problèmes de sécurité et de confidentialité entourant l'IoT à différentes couches. Dans ce contexte, nous proposons trois mécanismes de sécurité efficaces et robustes pour les systèmes IoT afin de contrecarrer les attaques du monde physique et du cyberespace. Nous évaluons les performances et la sécurité de nos solutions proposées en utilisant la logique Burrows-Abadi-Needham (BAN), l'outil de validation automatisée des protocoles et applications de sécurité Internet (AVISPA) et Network Simulator 3 (NS-3). Les résultats obtenus montrent que nos techniques proposées sont sécurisées, efficaces et adaptées aux systèmes IoT par rapport aux méthodes connexes récentes.
***Mots clés*** IoT, Réseaux de capteurs sans fils, Cryptographie sur les courbes elliptiques, Cryptage léger, Blockchain, Authentification, Confidentialité, Trust.

**الملخص**

إن انترنت الأشياء هي موجة ناشئة من الإنترنت جذبت الكثير من الاهتمام في السنوات الأخيرة. لقد أحدثت تغييرًا جذريًا في الحياة المجتمعية حيث يمكن للأشياء في إنترنت الأشياء أن تتخذ مجموعة متنوعة من الأشكال ، من الأشياء الصغيرة إلى الكبيرة. يواجه النشر الهائل لإنترنت الأشياء مشكلات تتعلق بالأمان والخصوصية ، والتي سيكون لها مخاطر كبيرة لتتماشى مع الفوائد المحتملة لإنترنت الأشياء. يثير تأمين مثل هذه الأنظمة العديد من التحديات لا سيما في البيئات ذات الموارد المحدودة وغير المتجانسة والواسعة النطاق. الهدف الرئيسي من هذه الأطروحة هو التغلب على مشكلات الأمان والخصوصية المحيطة بإنترنت الأشياء في طبقات مختلفة. في هذا السياق، نقترح ثلاثة أنظمة أمان فعالة وقوية لأنظمة إنترنت الأشياء لإحباط الهجمات من كل من العالم المادي والفضاء الإلكتروني. نقوم بتقييم أداء وأمن حلولنا المقترحة باستخدام منطق بان وأداة التحقق الآلي من بروتوكولات وتطبيقات أمان الإنترنت و محاكي الشبكة 3. تظهر النتائج التي تم الحصول عليها أن تقنياتنا المقترحة آمنة وفعالة ومناسبة لأنظمة إنترنت الأشياء مقارنة بالطرق الحديثة ذات الصلة.
***المفاتيح*** انترنت الأشياء، شبكة الاستشعار اللاسلكي، تشفير بالمنحنيات الاهليلجية، تشفير خفيف، بلوك تشين، المصادقة، السرية، الثقة.

# Acknowledgment

All praise is due to Almighty Allah, the compassionate and merciful, who enabled me to accomplish this work.

My sincere gratitude goes to my supervisor Pr. Allaoua Refoufi for his professional guidance and academic support.

I am extremely grateful to my co-supervisor Pr. Zibouda Aliouat for her careful remarks and enlightening help.

I would deeply acknowledge Pr. Saad Harous and Dr. Abdelhak Mourad Gueroui for his valuable advice, wisdom and humility.

I am also thankful to committee members of my dissertation for offering their time to review this document.

As I would like to express my special thanks of gratitude to all my teachers who helped me to reach this level.

# Dedication

*To my mother whose infinite love and great help encourage me to realize my dreams*

*To my father, I would like to thank him for his trust and love*

*To my sister, I would like to thank her for energizing me with positivity*

*To the most wonderful brother in the world "Anis"*

*To all my family and friends*

# Table of contents

# List of Figures

# List of Tables

# List of Algorithms

# List of Publications

- **A Review of Security in Internet of Things**

  *Authors:* Yasmine Harbi, Zibouda Aliouat, Saad Harous, Abdelhak Bentaleb, Allaoua Refoufi

  *Journal:* Wireless Personal Communications

  *Year:* 2019

  *Status:* Published

- **A survey on security of IoT : vulnerabilities, emerging solutions, challenges and future directions**

  *Authors:* Yasmine Harbi, Zibouda Aliouat, Allaoua Refoufi, Saad Harous

  *Journal:* Future Generation Computer Systems

  *Year:* 2020

  *Status:* Under review

- **Enhanced Authentication and Key Management Scheme for Securing Data Transmission in the Internet of Things**

  *Authors:* Yasmine Harbi, Zibouda Aliouat, Allaoua Refoufi, Saad Harous, Abdelhak Bentaleb

  *Journal:* Ad Hoc Networks

  *Year:* 2019

  *Status:* Published

- **Improved bio-inspired security scheme for privacy-preserving in the Internet of Things**

  *Authors:* Yasmine Harbi, Allaoua Refoufi, Zibouda Aliouat, Saad Harous

  *Journal:* Wireless Networks

  *Year:* 2020

  *Status:* Under review

- **Lightweight blockchain-based remote user authentication for fog-enabled IoT deployment**

  *Authors:* Yasmine Harbi, Allaoua Refoufi, Zibouda Aliouat, Saad Harous, Abdelhak Mourad Gueroui

  *Journal:* Journal of Network and Computer Applications

  *Year:* 2020

  *Status:* Under review

- **Secure data transmission scheme based on elliptic curve cryptography for Internet of Things**

  *Authors:* Yasmine Harbi, Zibouda Aliouat, Saad Harous, Abdelhak Bentaleb

  *Conference:* International Symposium on Modelling and Implementation of Complex Systems (Springer)

  *Location:* Laghouat, Algeria

  *Year:* 2018

  *Status:* Published

- **Efficient End-to-End Security Scheme for Privacy-Preserving in IoT**

  *Authors:* Yasmine Harbi, Allaoua Refoufi, Zibouda Aliouat, Saad Harous

  *Conference:* International Conference on Networking and Advanced Systems (IEEE)

  *Location:* Annaba, Algeria

  *Year:* 2019

  *Status:* Published

# General introduction

Internet of Things (IoT) is an emerging technology that has drawn a lot of attention in recent years. It refers to a broad vision whereby "things" such as everyday objects, places and environments are interconnected with one another via the Internet. IoT has made dramatic change of community life where things in IoT can take a wide variety of forms, from simple tags attached to merchandises, smart thermostats installed in the classrooms, implantable medical devices on the patients, to video cameras on top of light poles, and automobiles with built-in sensors. The explosive deployment of IoT has pushed the boundary of the cyber-world to be tightly intertwined with our physical world [1, 2].

The IoT enables the exchange of information in a variety of application scenarios, each having unique characteristics and requiring unique performance guarantees, and together they bring potentially tremendous benefits to human being, such as: home automation, environmental monitoring, health and lifestyle, smart cities, etc.

Nowadays, the emergence of Cloud Computing has created the application and device management backbone needed to scale to and support billions of connected things. In the last white paper IoT Cisco report, IoT will account for an increasingly huge number of connections; 50 billion devices by 2020 [3].

The IoT is the result of the development and combination of different technologies. It encompasses different existing concepts such as wireless sensor networks (WSNs) and radio frequency identification (RFID), and uses advanced technologies such as Cloud Computing, Big data, or blockchains [4].

WSNs play a vital role in IoT environments because they cover several applications including healthcare, industry and agriculture domains. Sensors are generally resource-

constrained; they have limited energy, processing and storage capacities.

The IoT provides advanced services such as real-time monitoring, control management and automation processes. Therefore, it brings a lot of economic gains to the suppliers in particular, and society in general. It has inspired thousands of researchers and developers around the world to develop and improve this growing infrastructure. Several IoT-based platforms mainly use communication technologies with low energy consumption where objects are generally limited in energy, processing and storage (*e.g.* sensors, smartphones, drones, etc).

## Problem statement

Th growth of IoT is faced by security and privacy issues, which will have significant risks to go along with the potential benefits of the IoT. For instance, as we add devices to our cloths, bodies, homes, and environments, more personal information will be collected.

As devices are more closely connected with our physical world and some are capable of taking actions, data security, device security and communication security become critically important. In 2016, Anna Senpai created a malicious program, called Mirai, which is possible to take control of vulnerable connected objects such as surveillance cameras and routers, and generate distributed denial of service attacks (DDoS). Mirai transforms the infected objects into autonomous and intelligent agents that are controlled remotely [5].

The proliferation of IoT can be achieved by providing a good level of security. It is very necessary to design new security frameworks that prevent any malicious or unauthorized object from gaining access to IoT systems, read or modify the collected data. Connected objects must be authenticated before joining the network or exchanging any information. However, they have generally very limited resources, and thus conventional security mechanisms such as Rivest Shamir Adleman (RSA) can not be supported by the objects. Therefore, a new security mechanisms must be developed to provide authentication, confidentiality and privacy, while being suitable for constrained IoT devices [6–8].

A secure and trustworthy IoT is a hard and complex task especially in resource-

constrained, heterogeneous and large-scale environments. It demands multiple lines of defense from different layers to thwart attacks from both the physical world and cyberspace.

## Goals and contributions

The main objective of this thesis is to research the security and privacy related issues surrounding the IoT and develop new security mechanisms for the IoT. The proposed approaches must take into consideration the limited resources of IoT devices and address limitations of previous methods to improve the IoT security. To overcome such problem, we opted for WSNs-based IoT environments; because WSNs are deployed in various IoT domains (*e.g.* industrial, environmental, medical, etc). Initially, we introduce common elements and protocols of IoT to demystify the origins of threats in IoT. We provide a taxonomy of IoT attacks and analyze the security threats of IoT at diffeerent layers. We also present a taxonomy of the security requirements based on the attacks' purposes. Then, we propose three solutions to improve the IoT security at different layers; perception, network and application. The first scheme is based on elliptic curve cryptography (ECC) and aims to verify the identity of the communicating objects before negotiating a cryptographic key. The second scheme achieves privacy-preserving in WSN-based IoT, it is based on lightweight operations to encrypt and decrypt the collected data using a symmetric shared key. Since the IoT sensors have limited storage capability, the encrypted data is stored on a cloud server. The third scheme enables end-user devices to access data in the cloud after successful authentication and session key agreement using blockchain technology.

## Dissertation outline

In this thesis, all chapters are transcription of our articles published in or submitted to scientific journals. It contains six chapter divided equally into two main parts: background and contributions. The background part gives an overview of IoT and discusses its security threats, requirements, solutions and challenges. The contributions part presents our proposed techniques that improve the IoT security. This dissertation is organized as follows:

- Chapter 1 defines the concept of IoT, presents the enabling technologies that motivate the emergence of IoT, and introduces common applications and elements of IoT.
- Chapter 2 analyzes the IoT attacks and discusses different security requirements that must be achieved to overcome such threats. It also provides a taxonomy of IoT attacks and a classification of IoT security requirements.
- Chapter 3 introduces emerging solutions that can improve the IoT security and presents the state-of-the-art on security algorithms for IoT. It also discusses the security challenges related to the proposed solutions.
- Chapter 4 presents an enhanced authentication and session key agreement for WSNs-based IoT using ECC. The security of the enhanced scheme is formally verified using the Burrows-Abadi-Needham logic and the Automated Validation of Internet Security Protocols and Applications tool.
- Chapter 5 proposes a bio-inspired security scheme to achieve privacy-preserving in the IoT. The improved scheme is based on a genetic algorithm and a chaotic system to encrypt/decrypt multimedia data. We formally verified the security of the proposed scheme using AVISPA tool and evaluate its performance using NS-3.
- Chapter 6 introduces a lightweight and distributed multifactors remote user authentication scheme for IoT. The proposed scheme is based on blockchain technology and fog computing and uses a lightweight cryptographic hash function. The proposed scheme is formally verified using the widely-accepted AVISPA tool and implemented using solidity language.

# Background

**Chapter 1:** Internet of Things : an overview

**Chapter 2:** Review of security in Internet of Things

**Chapter 3:** IoT security solutions and challenges

# Chapter 1

# Internet of Things : an overview

## 1.1 Introduction

Over the past few years, the IoT has gained significant attention since it brings potentially tremendous benefits to the human. The concept of the IoT has been introduced by Kevin Ashton in 1999, it aims to connect anything at anytime in anyplace [1]. "Things" in IoT are embedded with sensing, processing and actuating capabilities and cooperate with each other to provide smart and innovative services autonomously.

The IoT spans many diverse application domains such as home automation, environmental monitoring, healthcare, and so on [2]. The primary objective of the IoT is unification of these numerous diverse application domains under the same umbrella referred as smart life [2].

The architecture of IoT supports a large number of heterogenuous devices and integrates various communication technologies that enable the connectivity of IoT devices to provide the required services to end-users.

The present chapter provides an overview of fundamental concepts of IoT. It introduces the IoT definition, potential applications and architecture including major elements and protocols used in IoT.

## 1.2 IoT definition

The Internet of Things (IoT) refers to a growing network of everyday physical objects connected to the Internet. It allows the transformation of Internet-enabled devices

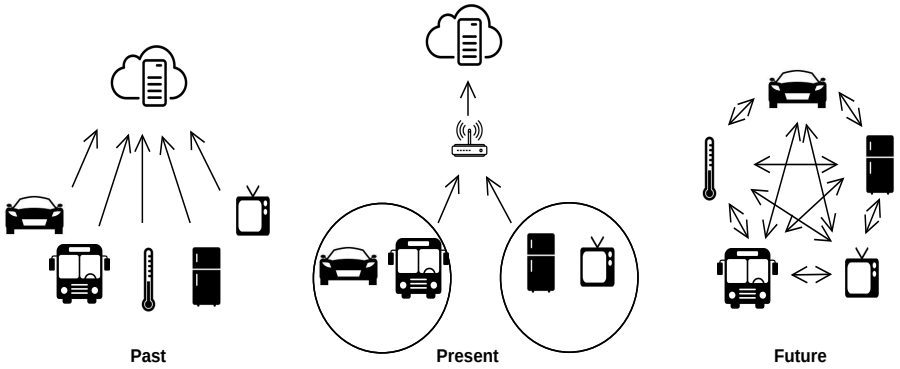


Figure 1.1: Evolution of IoT.

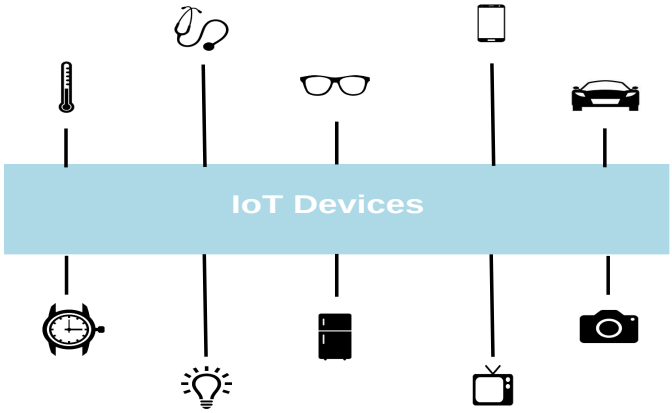


Figure 1.2: Examples of IoT devices.

to an interconnected ecosystem with digital data accessible anywhere and anytime. The IoT devices include physical objects ranging from tiny to large machines that seamlessly communicate with each other via the Internet without human intervention [9]. Figure 1.1 demonstrates the evolution of IoT where devices will be connected to each other and exchange data through the Internet.

According to Cisco, 50 billions of devices are currently estimated to be connected to the Internet [3]. The IoT devices are equipped with sensors to smartly perceive their surroundings and actuators to autonomously perform actions [10]. Figure 1.2 highlights various examples of the IoT devices. These devices are inherently resource-constrained, they have limited memory space, low processing capacity and computation power.

Different enabling technologies such as wireless sensor networks (WSNs), radio frequency identification (RFID) and cloud computing evolve as an essential components for the emergence of IoT paradigm [11]. Some available technologies are illustrated in Figure 1.3.

— Wireless Sensor Network (WSN) consists of a large number of physical autonomous

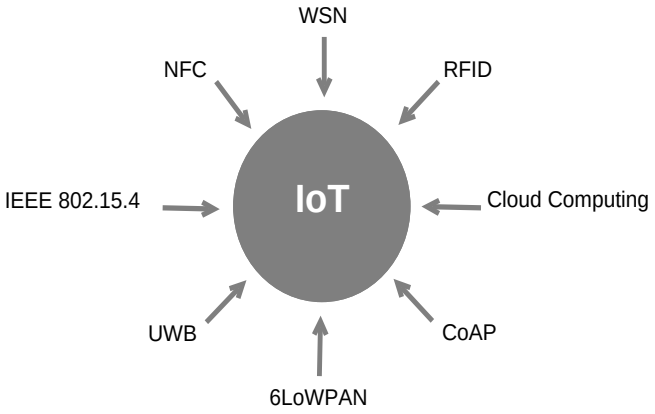


Figure 1.3: IoT enabling technologies.

sensors deployed in the environment in order to control the environmental conditions [1].

— Radio Frequency IDentifiaction (RFID) is used to identify and track IoT objects. It allows data exchange via radio signals over a short distance [1].

— Cloud computing plays an important role in the IoT by offering an unlimited storage ressources and processing power [12].

— Constrained Application Protocol (CoAP) is an application layer protocol proposed for ressource-constrained devices [4] [13].

— IPv6 Low power Wireless Personal Area Network (6LoWPAN) combines IPv6 and LoWPAN and allows transmission of IPv6 packets over IEEE 802.15.4 networks [4]. The 6LoWPAN is suitable for the IoT and has several advantages.

— Ultra WideBand (UWB) is a viable technology for a wide variety of IoT applications due to its low power consumption, higher precision, and security [14].

— IEEE 802.15.4 is a protocol for the physical layer and the MAC (Medium Access Control) layer in Wireless Personnal Area Networks (WPANs). It provides the connection of things in personal area with low energy consumption [4].

— Near Field Communication (NFC) is a short range technology that can be used in various IoT systems such as payments and authentication. The NFC provides easy network access and data exchange [15].

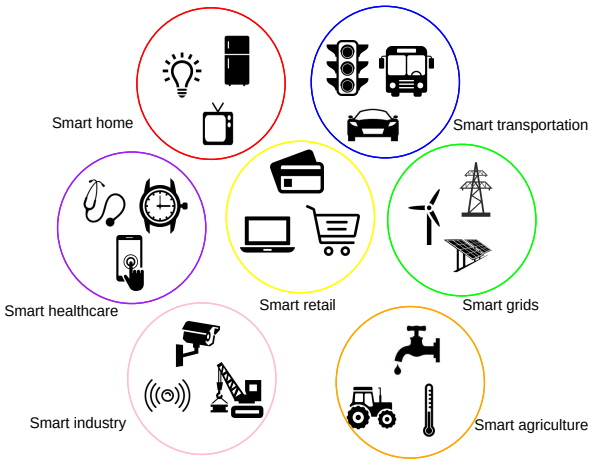


Figure 1.4: IoT applications.

## 1.3 IoT applications

The IoT provides a large number of applications to enhance peoples' daily lives and activities. Figure 1.4 shows potential examples of IoT applications.

**Smart home** encompasses a collection of smart devices (*e.g.,* smart lock, baby monitor, fire detector) deployed at home and locally communicate over wireless channels. Home devices can be remotely accessed through a home gateway.

**Smart healthcare** enables collection, transmission and storage of patients' physiological information. For instance, patient's heart rate can be collected by medical sensors and transmitted to hospital server for diagnosis and tracking purposes.

**Smart transportation** includes a large number of smart vehicles which can communicate with each other (vehicle-to-vehicle), to outside station (vehicle-to-infrastructure) and to pedestrians (vehicle-to-pedestrian) over wireless networks. A smart vehicle can detect current traffic status, manage speed, and exchange data to provide efficient and safe driving.

**Smart agriculture** allows remote control of temperature, humidity, irrigation, soil moisture and micro-climate conditions to provide high production/quality and prevent financial losses. In an intelligent farming system, sensors can be attached to animals to track livestock behaviors and health conditions.

**Smart industry**, known as industrial IoT (IIoT) uses machine-to-machine technology to automate the process of manufacturing with insignificant human intervention. The IIoT aims to better control the production process, data, and issues to provide efficient and reliable final products.

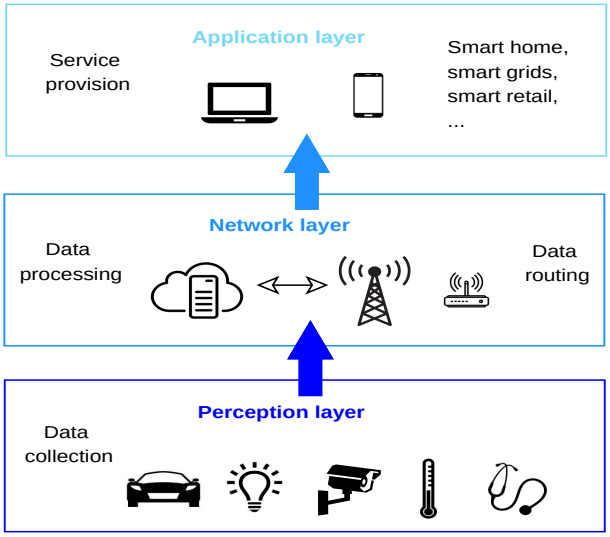


Figure 1.5: Three-layered IoT architecture.

**Smart retail** permits the tracking of products in warehouses or during traveling. Sensors can be attached to a retail item to track the product status. Various smart shopping systems were developed to provide intelligent services for customers and thus gain more clients.

**Smart grid** is a common application of IoT that measures, monitors, and manages electricity consumption. It enables efficient and reliable electricity management, provides energy saving and reduces powers grids issues/failures.

## 1.4 IoT architecture, elements and protocols

The architecture of IoT is not standardized, typical IoT architecture has three layers: perception, network and application [16] as shown in Figure 1.5.

### 1.4.1 Perception layer

The perception layer includes different physical IoT devices; it is responsible for interaction among devices and collection of IoT data. Data collection is performed using smart devices such as radio frequency identification (RFID) tags and sensors.

#### 1.4.1.1 Wireless sensors

Wireless sensors play an essential role in IoT by providing sensing and communicating services [17]. A Wireless sensor network (WSN) consists of a large number of

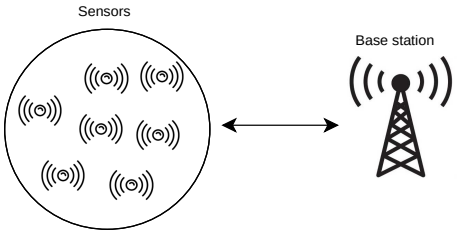


Figure 1.6: WSN architecture.


Figure 1.7: RFID system.

intelligent sensors deployed in remote environments to sense and collect data such as temperature, humidity, vibration, etc. Sensed data are transmitted through one or multi-hop to a gateway/base station as depicted in Figure 1.6.

#### 1.4.1.2 Radio frequency identification (RFID)

RFID technology is a major element of IoT due to its identification, tracking and monitoring of objects [18]. An RFID system consists of radio signal transponder (tag) that stores a unique identity of object and a tag reader that identifies the object through radio waves. The tag reader transfers the identification number to a computer to track and monitor the object as shown in Figure 1.7.

### 1.4.2 Network layer

The network layer processes the collected data provided by the perception layer and stores or sends the data to the application layer. It is the most important layer of IoT architecture because it integrates various communication technologies that enable the connectivity of IoT devices. The widely used communication technologies include ZigBee, Bluetooth low energy (BLE), IPv6 over low power wireless personal area networks (6LoWPAN) and long range wide area network (LoRaWAN). Table 6.3 provides a comparison of the studied IoT wireless technologies. This comparison helps to select the suitable protocol for a defined IoT system.

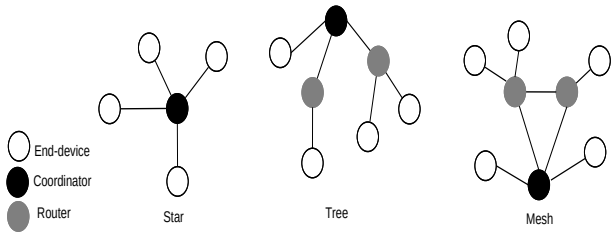


Figure 1.8: ZigBee topologies.

#### 1.4.2.1 ZigBee

ZigBee is a wireless communication technology designed for short-range communications [19]. It can be used in smart homes, smart meters and smart healthcare. The ZigBee protocol stack includes physical (PHY) and medium access control (MAC) layers based on IEEE 802.15.4 standard [20], a network (NWK) layer and an application (APP) layer. A ZigBee network can have a star, tree or mesh topology and each network has a coordinator node (trusted node) that manages the network and maintains security between devices. In star network, end-devices are directly connected to the coordinator while in tree or mesh networks, intermediate routers are used to extend the network, as shown in Figure 1.8. The NWK layer provides data routing using cluster-tree and modified ad hoc on-demand distance vector (AODV) algorithms [21]. A ZigBee device can only communicate with another ZigBee device, and thus, it has limited interoperability.

#### 1.4.2.2 BLE

BLE is a short-range communication technology that reduces energy consumption compared to classic Bluetooth [22]. It is widely used in IoT vehicular systems. BLE has a protocol stack composed of PHY layer, MAC layer, logical link control and adaptation protocol (L2CAP) and attribute protocol (ATT). The BLE adopts a star topology including master and slave devices as demonstrated in Figure 1.9. Each slave node is associated with a single master node. The master node is responsible to initiate the communication and provide scheduling table according to time division multiple access (TDMA).

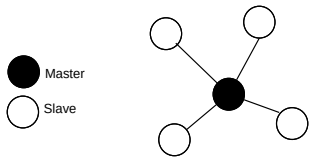


Figure 1.9: BLE topology.

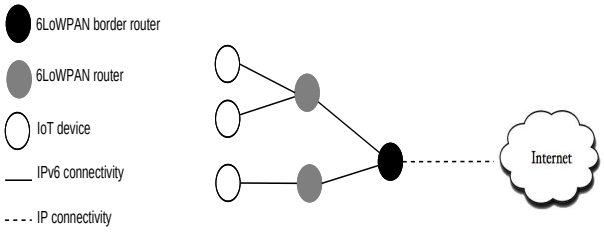


Figure 1.10: 6LoWPAN architecture.

#### 1.4.2.3 6LoWPAN

6LoWPAN combines the latest version of Internet protocol (IPv6) and low power wireless personal area network (LoWPAN) [23]. It enables IoT devices with limited capabilities to transmit data through wireless channels using IPv6. It is suitable for resource-constrained devices because it reduces transmission cost, supports mobility, etc. The most common use cases of 6LoWPAN are smart home, smart agriculture and industrial IoT. Compared to ZigBee, a 6LoWPAN device can communicate with another 6LoWPAN device or IEEE 802.15.4 device. It can also communicate with an IP-based network such as Wi-Fi as presented in Figure 1.10. The specification of 6LoWPAN defines a complete protocol stack that consists of PHY and MAC layers based on IEEE 802.15.4 standard, the NWK layer, the transport layer and APP layer [24].

The routing within 6LoWPAN network uses routing protocol for low-power and lossy networks (RPL) [25]. RPL supports point-to-point, point-to-multipoint and multipoint-to-point communications. It is based on direct acyclic graph (DAG). From DAG, RPL creates a destination oriented direct acyclic graph (DODAG) tree that contains one root from leaf node to the root.

#### 1.4.2.4 LoRaWAN

LoRaWAN is a long-range communication protocol designed for low power and scalable IoT applications [26]. As depicted in Figure 1.11, a LoRaWAN network consists of end-devices, gateways and a single server in a star or star-of-star topology. The end-

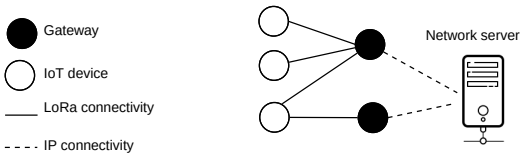


Figure 1.11: LoRaWAN architecture (star-of-star topology).

Table 1.1: Comparison of IoT wireless technologies.

| **Wireless technology** | ZigBee | BLE | 6LoWPAN | LoRaWAN |
|---|---|---|---|---|
| **Topology** | star, tree, mesh | Star | Star, mesh | star, star-of-star |
| **Range** | 10-20m | <100m | 10-20m | 3-5km |
| **Application** | smart home, smart meters, smart healthcare | smart vehicule | smart home, smart agriculture, smart industry | smart city |
| **Interoperability** | No | No | Yes | Yes |
| **Security** | Yes | Yes | No | Yes |
| **Scalability** | Yes | No | Yes | Yes |

devices can communicate to one or more gateways using ALOHA scheme through one-hop links. The gateways are connected to the network server via Internet protocol. The communications are bidirectional and initiated by the end-device.

### 1.4.3 Application layer

The application layer receives the data from the network layer and provides the required services to IoT users. It supports a large variety of applications such as smart home, smart retail, smart grids, etc. The most common application protocols are constrained application protocol (CoAP) and message queuing telemetry transport (MQTT). A comparison of these protocols is provided in Table 1.2.

#### 1.4.3.1 CoAP

Since IoT devices are resource-constrained, HTTP protocol is not suitable for low power devices due to its complexity. CoAP was designed to include features of HTTP dedicated to IoT devices. As demonstrated in Figure 1.12, CoAP is a messaging protocol based on representational state transfer (REST) architecture [27]. It has four message types: confirmable, non-confirmable, acknowledgment and reset. It provides features that are not available on HTTP such as push notification (*i.e.,* the server sends

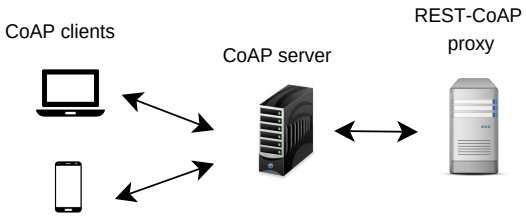


Figure 1.12: CoAP architecture.

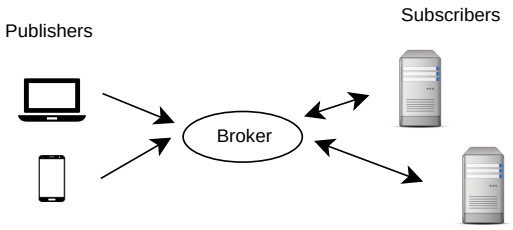


Figure 1.13: MQTT architecture.

Table 1.2: Comparison of IoT application protocols.

| **Application protocol** | CoAP | MQTT |
|---|---|---|
| **Transport layer** | UDP | TCP |
| **REST** | Yes | No |
| **Request/response** | Yes | No |
| **Publish/Subscribe** | Yes | Yes |
| **Security** | DTLS | SSL |

notification to the device) and resource discovery (*i.e.,* the server can store the list of devices).

#### 1.4.3.2 MQTT

MQTT is a lightweight messaging protocol that provides the connectivity of networks and users with applications. It is based on publish/subscribe architecture where the system consists of three main components: publishers, subscribers and a broker as presented in Figure 1.13. In the context of IoT, publishers are embedded devices that send data to the broker and subscribers are applications servers.

## 1.5 Conclusion

The IoT has drawn significant attention in recent years since it has made revolutionary changes in human life. The IoT enables the exchange of information in a wide variety of applications such as smart buildings, smart health, smart transport,

and so on. In this chapter, we introduced the definition of IoT network and presented enabling technologies that motivate to the emergence of IoT. Moreover, we reviewed different applications provided by the IoT paradigm and discussed the major elements and protocols integrated in the three-layered IoT architecture.

In next chapter, we focus on security vulnerabilities and requirements of IoT. We present different security attacks that threaten the IoT environments. We provide a valuable taxonomy to highlight the security threats of IoT. To achieve the desired level of security in IoT, we propose a new taxonomy of security requirements including data security, communication security and device security.

# Chapter 2

# Review of security in Internet of Things

## 2.1 Introduction

Despite the interesting growth of IoT and the emergence of potential services that will be offered to improve human lives, the IoT faces several issues. As the IoT combines different existing technologies such as WSN and RFID, it inherits the security flaws of each technology [6]. Moreover, billions of devices are expected to be connected to the Internet [7]. Hence, an increasingly massive amount of data will flow within the Internet [8]. This data can face various security attacks such as eavesdropping and altering. Consequently, the user's privacy will be threatened [28]. For example, an adversary can intercept a baby monitor system using a Software Defined Radio (SDR) in order to compromise the user's privacy [29].

Security is a major concern that inevitably affects the IoT networks. The security of IoT has attracted significant attention of researchers' community [30,31]. To develop a secure system, security vulnerabilities and attacks of IoT should be analyzed. Several security requirements and properties such as authentication, confidentiality, integrity, and so on must be ensured to secure IoT systems.

This chapter presents a deep analysis of the IoT security threats and requirements. It introduces different IoT security attacks, their definitions and purposes. A taxonomy of IoT attacks including levels, purposes and countermeasures is provided. A taxonomy of security requirements based on attacks' purposes is also presented.

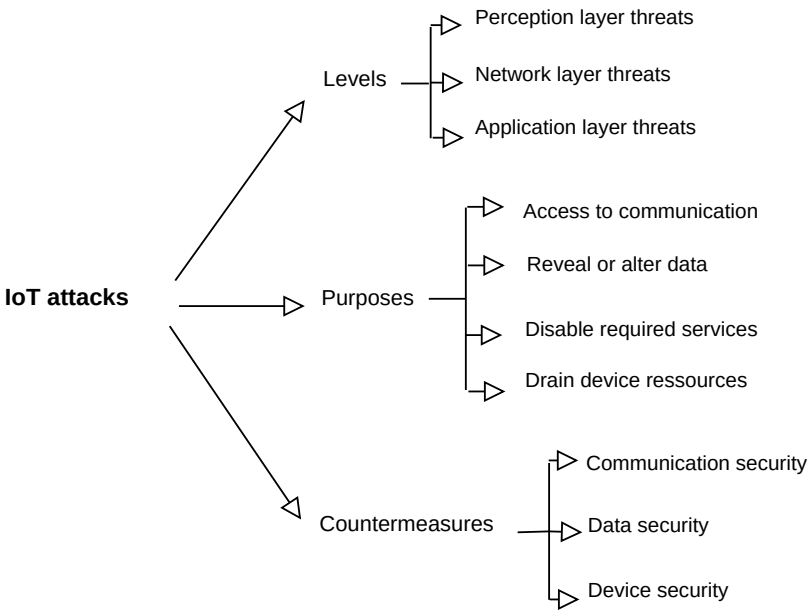


Figure 2.1: Taxonomy of IoT security attacks.

## 2.2 IoT security attacks

The IoT is evolving very fast, and the security attacks are advancing as well. To include the security requirements carefully into the IoT systems, it is firstly necessary to analyze the IoT vulnerabilities and attacks. The IoT is prone to various types of attacks since it combines different existing technologies such as WSN and RFID. Therefore, the IoT inherits the security flaws of each technology. Table 2.1 provides different security attacks that threaten the IoT networks.

## 2.3 IoT security threats

According to the studied security attacks, we provide a taxonomy of IoT attacks based on levels, purposes and countermeasures as shown in Figure 2.1. In this section, we focus on the security vulnerabilities of IoT at the three layers.

**Levels** examine the security issues of IoT at the three layers. Perception layer threats address the security attacks within major elements of IoT such as WSNs and RFID. Network layer threats analyze vulnerabilities of the aforementioned communication protocols. Application layer threats include attacks related to IoT software and end-user devices.

**Purposes** evaluate the impacts of security attacks on IoT systems. The main

Table 2.1: IoT security attacks.

| **Attack** | **Description** | **Purposes** |
|---|---|---|
| Node tampering [32] | Replace physically the sensor node or part of its hardware | Access to sensitive information and affect the services' availability |
| Node injection [6] | Deploy physically malicious nodes in the IoT network | Control data flow, access to private information and launch additionnal attacks |
| Node capture attack [33] | Capture node from the network | Obtain sensitive information |
| Black hole attack [34] | Send route replay messages to source node to receive packets from the sender node | Access to private data and join the network |
| Sinkhole attack [35] | Claim unconstrained capabilities to be selected for forwarding all traffic in WSN | Breach the data confidentiality and launch additional attacks |
| Wormhole attack [36] | Create a false one-hop transmission (tunnel) to deliver more data through this tunnel | Breach the data confidentiality and launch additional attacks |
| Sybil attack [37] | Pretend the identities of many other nodes to be in more than one location | Degrade the data security and resource utilization |
| Sleep deprivation [6] | Break the programmed sleep routines of IoT devices and keep the sensor nodes awake all times | Drain the devices' resources and make the IoT system dysfunctional |
| RFID Sniffing [38] | Sniff out or eavesdrop the data flow in RFID system using various sniffing applications | Disclosure sensitive data |
| RFID Spoofing [6] | Spoof or imitate valid RFID information and send data with the valid tag ID to mask the attacker identity | Elicit sensitive data and gain access to the system |
| RFID Cloning [6] | Clone an RFID tag by duplicating data from a pre-existing RFID tag | Gain access to the system |

| **Attack** | **Description** | **Purposes** |
|---|---|---|
| Replay attack [32] | Eavesdrop the communication and retransmit the packets to destination node | Obtain the confidence and trust of the IoT system and launch additinal attacks |
| Man in the middle (MTM) [39] | Intercept and possibly alter the communication between two nodes | Get private information and launch additinal attacks |
| Eavesdropping attack [39] | It is a subset of MTM where an attacker intercepts secretly the communications | Get private information |
| Brute force attack [40] | Try many keys to guess the correct one | Decrypt encrypted data |
| Encryption attack [6] | Use particular techniques like timing, power, fault and electromagnetic analysis on IoT devices to find the encryption key | Break the encrypted system and get private data |
| Code injection [6] | Inject malicious code that will be executed by the IoT system | Control the whole system and compromise data integrity, privacy and correctness |
| Denial of Service (DoS) [40] | Send many packets to the IoT system | Exhaust the service provider resources, disable the network and compromise data acquisition |
| Node jamming [41] | It is a subset of DoS where the attacker can jam the signals by sending noise | Make intentional interferences in the network and disable the network |
| Phishing attack [42] | Trick IoT devices or users usually by disguising as trustworthy entity | Gain access to sensitive information |
| Social engineering [6] | Manipulate or influence the users of the IoT system based on human interaction | Access to confidential information |
| Malicious software [6] | Infect or cripple an IoT system with malicious software like virus, worms, trojan horse, *etc* | Damage connected IoT devices and components, tamper and steal information |

purposes of IoT attacks are the followings:

— Access to communication.

— Reveal or alter data.

— Disable required services.

— Drain device ressources.

**Countermeasures** consist of the security requirements to mitigate the identified purposes of IoT attacks. This class includes communication security, data security and device security. IoT communications can be secured by providing authentication, access control and non-repudiation. To protect data, relevant security requirements such as confidentiality, privacy and integrity must be considered. Other fundamental requirements including trust and availability of IoT devices are needed in different environments.

### 2.3.1 Perception layer threats

The limited resources and heterogeneous nature of IoT devices make them vulnerable to various security attacks.

WSNs are generally deployed in harsh and unattended environments and thus they are prone to several attacks. Common security attacks of WSNs are sinkhole, blackhole, wormhole, sybil, denial of service (DoS), node capture, and node injection attack [43]. Brief descriptions of these security attacks are provided in Table 2.1.

Similar to the WSN, the RFID networks are susceptible to different type of attacks including spoofing, cloning, and sniffing attack (See Table 2.1).

The IoT inherits the security threats of WSNs and RFID because they are vital elements of IoT networks.

### 2.3.2 Network layer threats

ZigBee protocol implements security mechanisms including advanced encryption standard with cipher block chaining message authentication code (AES-CCM) and message integrity code (MIC) to provide confidentiality, authentication and integrity. The ZigBee security is based on three keys: a link key (for unicast communications), a network key (for broadcast communications) and a master key (for link key and network

key generation). As mentioned in [44], the master key is installed in the device during manufacturing process. The link key can be generated using key transport or key establishment methods, while the network key can be acquired using key transport method.

As the master key is stored on the device, an attacker can read it from the memory after the success of node capture attack. Another possible attack presented in [45] that aims to drain energy of ZigBee nodes. The authors in [46] evaluated the vulnerability of ZigBee network against sinkhole attack. In [47], the authors showed that three ZigBee-based smart light systems are unsecure to several types of attacks such as denial of service (DoS), network key extraction and code injection attacks.

BLE protocol provides confidentiality and authentication using 128-bits AES-CCM algorithm as ZigBee. The symmetric key is generated using pairing procedure. First, the IoT devices exchange necessary information for authentication. Second, they generate and exchange temporary keys based on a pairing method. Finally, the device may exchange and store common keys to be used for further communications.

The pairing methods have several security issues including eavesdropping, man-in-the-middle (MTM) and brute force attacks as presented in [48] and [49]. Latter, new pairing procedure has been designed based on elliptic curve diffie hellman (ECDH). However, the authors in [50,51] demonstrated that it has similar problems. In [52], the authors presented another type of attacks such as data leackage and DoS attack that can be performed in a BLE-based smart door lock system.

6LoWPAN protocol enables resource-constrained devices to connect to the Internet using IPv6 addresses. It uses IPv6 header compression and packet fragmentation to reduce transmission overhead. However, it does not provide confidentiality, authentication or integrity preservation. An adversary can inject fake fragments with the header of a legitimate fragment; the receiver node uses the injected fragment in packet reassembly which causes the construction of a corrupted packet. Consequently, the buffer space of the receiver node will be reserved and not be able to receive further fragments [53]. Consecutive repetitions of fragment injection attack lead to a DoS attack [54].

RPL defines three security modes: unsecured, preinstalled and authenticated in the packet header. The unsecure mode is adopted when security is provided by MAC

Table 2.2: Security threats of IoT wireless technologies.

| **Wireless technology** | **Security attacks** |
|---|---|
| ZigBee | Encryption key, sinkhole, DoS, code injection |
| BLE | Eavesdropping, MTM, DoS, brute force |
| 6LoWPAN | Fragment injection, sinkhole, blackhole, sybil, DoS |
| LoRaWAN | Encryption key, DoS, MTM |

layer. In preinstalled mode, preinstalled keys are used to join the RPL network. The authenticated mode is not fully defined by the specification of RPL. If security is not provided at any layer, an attacker can perform different types of attacks in RPL network. A sinkhole, blackhole, flooding, sybil and DoS attacks against RPL networks are presented in [54–56].

The security of 6LoWPAN relies on securing communications at the MAC layer or APP layer. The security of MAC layer is provided using AES-CCM and MIC. However, the specification of IEEE 802.15.4 does not define the key management procedure.

LoRaWAN protocol adopts 128-bits AES algorithm and MIC to guarantee data confidentiality and integrity. When an IoT device is allowed to join the LoRaWAN network, the network server sends two session keys, namely network session key and application session key, to the end-device. These keys are used for data encryption/decryption and MIC. The main security weakness of LoRaWAN protocol is related to key management; an intruder can access to session keys using side channels attack since they are stored on the end-device. Moreover, the end-devices share the same session keys to secure multicast communications. This enables the intruder to read the keys from one node and thus reveal communications of other devices [57]. The authors in [58] demonstrated that LoRaWAN network is vulnerable to DoS and MTM attacks.

Table 2.2 summarizes the security threats of IoT communication protocols.

### 2.3.3 Application layer threats

CoAP is the application layer protocol that enables resource-constrained devices to achieve RESTful interactions. Since CoAP is built on UDP transport protocol, datagram TLS (DTLS) was proposed to provide confidentiality, authentication and integrity preservation in CoAP protocol [30]. However, limitations of DTLS can be considered as security threats of CoAP protocol [59].

Secure socket layer (SSL) was introduced to secure data transfer using MQTT

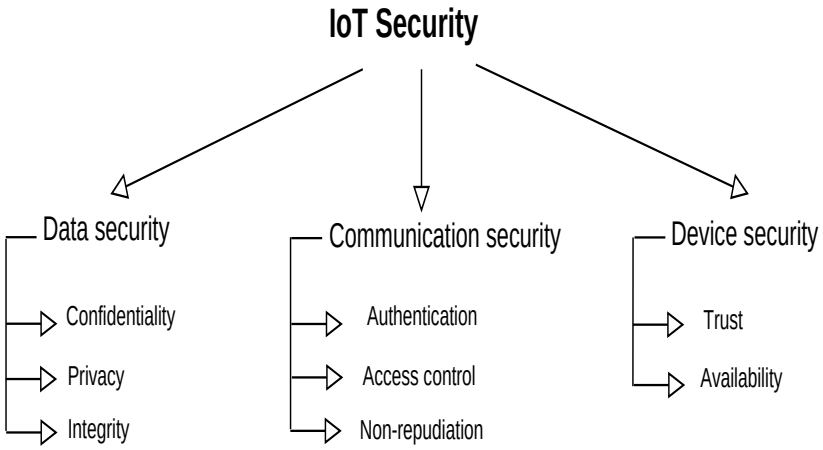


Figure 2.2: Taxonomy of IoT security requirements.

protocol. SSL uses asymmetric cryptographic technique to encrypt/decrypt the data. However, it stills prone to MTM attack [60]. An extension of MQTT called secure MQTT (SMQTT) was proposed to provide security during data transfer [61]. The publishers and subscribers register to the broker and get a secret key. This key is used for data encryption and decryption performed by publishers and subscribers, respectively. However, the key generation and encryption algorithms are not standardized.

In IoT, software vulnerabilities and users devices can be exploited by attackers. An adversary can impersonate or manipulate legal users to gain access to IoT system by injecting malicious software. The lack of user authentication has led to several IoT attacks such as Bashlite and Mirai attacks [5].

## 2.4 IoT security requirements

According to the IoT attacks' purposes demontrated in Figure 2.1, we classify the IoT security requirements into three categories: data security, communication security, and device security as depicted in Figure 2.2. To preserve sensitive information, we have to secure the data collected by the IoT devices. Also, we must secure the communication between these devices to avoid controlling the data flow by an adversary. In some IoT environments, the physical objects communicate with each other to provide intelligent services for human. Therefore, it is highly important to secure these devices.

### 2.4.1 Data security

The IoT devices monitor the physical environments and transmit the collected data through wireless channels. However, this transmitted data is exposed to different security threats like eavesdropping and altering. To secure data in the context of IoT, we must preserve its confidentiality, privacy, and integrity.

Data confidentiality is the process of hiding private information from the unauthorized IoT objects [62] [63]. According to [64], data confidentiality is a fundamental issue that needs a lot of attention. Standard encryption mechanisms cannot be implemented directly for the IoT system since IoT devices have limited resources [65]. In order to provide data protection and confidentiality, the authors in [66] proposed the use of lightweight cryptographic algorithms. The authors in [67] indicated that applying privacy-based designs can increase the confidentiality levels.

Privacy includes the concealment of personal information and the ability to control what happens with such information [68]. Data privacy must be analyzed during data collection, transmission, and storage. Many practical solutions have been proposed to deal with data privacy. These techniques include anonymization-based solutions, pseudo-random number generators, block ciphers, and stream ciphers [69].

Data integrity ensures that the data to be received has not been altered or modified during transmission [62]. Integrity involves maintaining the consistency, accuracy, and trustworthiness of the data. Several cryptographic hash algorithms (*e.g.* MD5 and SHA1) are used to ensure data integrity. However, most of these mechanisms cannot be implemented because the IoT devices are inherently resource constrained [70]. To address this concern, various lightweight hash functions were proposed like [71] and [72]. In case of detection of tampering data, error correction mechanisms such as Cyclic Redundancy Checks (CRC) and checksum functions can be used to solve the problem [6].

### 2.4.2 Communication security

Before any communication between IoT devices, an authentication process is required. Thus, only authorized devices can access to systems or information. Moreover, non-repudiation in the communication is achieved.

Authentication is the process of validating an identity using login and other informa-

tion like password, PIN and digital certificates [69]. It is required between two or group of parties in order to secure communication in IoT system. The authentication ensures that only authorized users can access the IoT devices and achieves non-repudiation in communications. When a new device is connected to the network, it should authenticate itself before exchanging data. The authentication can be verified using lightweight cryptographic algorithms, physical primitives, or biometric identification [73] [74].

Access control is a security feature that verifies the permission granted to users and systems to perform operations on other systems and resources [75]. The authors in [69] divided access control algorithms in five distinct types: role-based, organization-based, capability-based, attribute-based, and trust-based algorithms.

In the context of the IoT, non-repudiation is an essential element of network security [76]. It is the ability to ensure that an IoT node cannot repudiate having sent a message and that the receiver cannot deny having received the message [62]. The non-repudiation is particularly important in the business field (*e.g.* for digital contracts). It ensures that communications between two parties are valid and authentic. It can be achieved using Public Key Cryptography (PKC) [77].

### 2.4.3 Device security

To provide security in a critical environment, ensuring trust and confidence between interacting nodes is a primordial task. Furthermore, the availability of the IoT devices is highly required.

Trust is crucial for IoT users as stated in [78]. Trust management is the process of making decisions about communication with unknown entities [79]. In order to secure IoT system, it is necessary to interact with trusted IoT devices in order to prevent unwanted actions conducted by malicious nodes. According to [69], the trust management techniques are divided into two main categories: deterministic and non-deterministic trust. The deterministic trust encompasses policy-based and certificate-based mechanisms, while the non-deterministic trust includes recommendation-based, reputation-based, prediction-based, and social network-based systems.

The policy-based mechanisms use a set of policies to identify trust. In certificate-based approachs, trust is determined using public or private keys and digital signatures.

The recommendation-based systems utilize prior information to define trust. If there

is no prior information, the prediction-based methods can be used.
The reputation-based systems employs global reputation of entities, while the social network-based ones consider the entities' social reputation.

Device availability is an important factor in IoT systems since they can be utilized in crucial areas including economy, industry, healthcare, *etc* [80]. According to [4], the availability of IoT networks should be performed in hardware and software. Hardware availability of the IoT application means the existence of all devices all the time, while software availability is the ability of providing services anywhere and anytime.
The IoT devices may face several attacks such as DoS and DDoS that can hinder the services provided or affect the network availability. In [81], the authors emphasize the importance of detection and recovery of DoS/DDoS attacks in IoT environments.

## 2.5 Conclusion

The IoT refers to the next generation of the Internet. Shortly, billions of devices will be connected to the Internet. The IoT devices can exchange sensitive information that may be leaked. Hence, strengthening the IoT security is a major concern. In this chapter, we analyzed the security vulnerabilities and threats of IoT networks and provided a taxonomy of IoT attacks. To achieve a secure IoT system, we proposed a taxonomy of security requirements based on different attacks' purposes.

In next chapter, we present emerging solutions that have been proposed to achieve security requirements in IoT. We also review related work that address the security of IoT systems. Then, we discuss security challenges related to the emerging solutions.

# Chapter 3

# IoT security solutions and challenges

## 3.1 Introduction

The number of IoT devices and the variety of IoT applications have rapidly increased in last years. This growth faces several security issues that must be addressed [82, 83]. Several emerging technologies and approaches, namely fog computing, software defined networking, blockchain, lightweight cryptography, homomorphic and searchable encryption, and machine learning were introduced to increase the level of security in IoT.

The IoT networks are deployed on large scale and support heterogenuous devices. Most of IoT devices are resources-constrained, thus security-enhancing solutions must be computationally efficient. The emerging security solutions cannot be fully integrated to IoT systems because of the dynamic and heterogenuous nature and limited capalities of IoT devices. Consequently, these solutions impose different security challenges that need to be properly solved. It is challenging to trade-off between security and efficiency in IoT networks [43].

This chapter introduces emerging solutions that were proposed to improve the security of IoT systems. It reviews related work that address security of IoT applications at different layers. Moreover, it discusses the security challenges related to the emerging technologies and approaches.

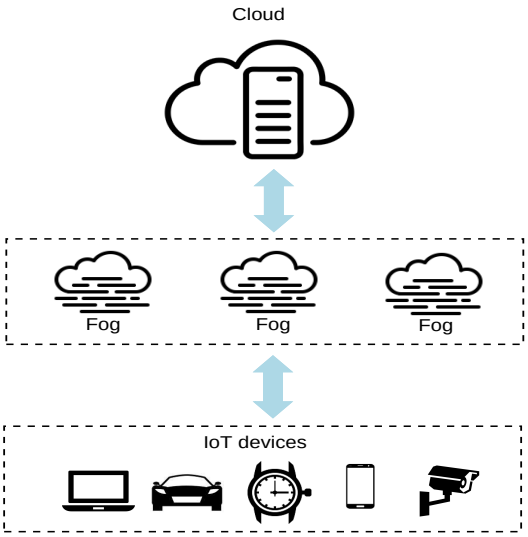


Figure 3.1: Fog computing architecture.

## 3.2 IoT security solutions

### 3.2.1 Fog computing-based solutions

Fog computing has been introduced as a new paradigm to extend (not to replace) the computational resources of Cloud computing. It provides storage, computation and networking/communication at the edge of the network [84]. Fog computing architecture consists of fog nodes deployed near to IoT devices and connected to the cloud server as shown in Figure 3.1.

The fog architecture helps to reduce the amount of data exchanged between the IoT devices and the cloud infrastructure. Fog computing supports mobility, location awareness, low latency, heterogeneity, scalability and thus can be perfectly adopted into real-time or latency-sensitive IoT applications. Since IoT devices have limited resources, fog nodes can provide various security requirements such as authentication [85–87], privacy-preserving [88–90] and encryption [91–93] to secure IoT environments.

### 3.2.2 Software defined networking-based solutions

Software defined networking (SDN) is an emerging computing concept that facilitates the network management by separating routing decisions of network elements (*e.g.,* routers, switches and gateways) and forwarding process. In SDN architecture, the network control operations like forwarding tables and ACL rules are handled by a centralized component called SDN controller, while data forwarding is managed by the

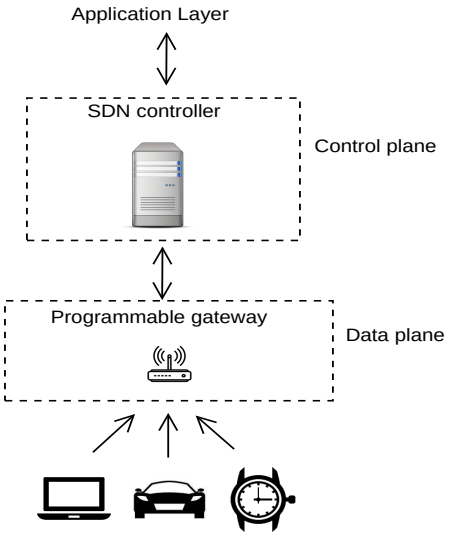


Figure 3.2: Software defined networking architecture.

network elements as depicted in Figure 3.2 [94].

The SDN can be an effictive solution for achieving security including key management [95], identity management [96], authentication [97, 98], confidentiality [99], and intrusion detection and mitigation [100, 101] in IoT applications .

### 3.2.3 Blockchain-based solutions

Blockchain is a disruptive technology that has revolutionized the world of cryptocurrency. It is a distributed ledger/database that contains transactions of nodes in a peer-to-peer (P2P) network [102]. A set of transactions are grouped into a single block and validated in a distributed way using a consensus algorithm. These blocks are linked to form a chain of blocks or blockchain as demontrated in Figure 3.5. Each block consists of two parts, the first part represents the validated transactions and the second part contains block timestamp, nonce value, hash of the block and hash of previous block.

The consensus process is executed by some nodes in the network called miners. Common consensus algorithms include power of work (PoW), power of stake (PoS), and practical byzantine fault tolerance (PBFT) [102]. These algorithms can be used to enable miners nodes to agree on adding a new block to the blockchain. The process of transaction's validation using blockchain is illustrated in Figure 3.4.

There are two main types of blockchain, namely public (permissionless) and private (permissioned) [102]. In public blockchain, any node can join the network while the

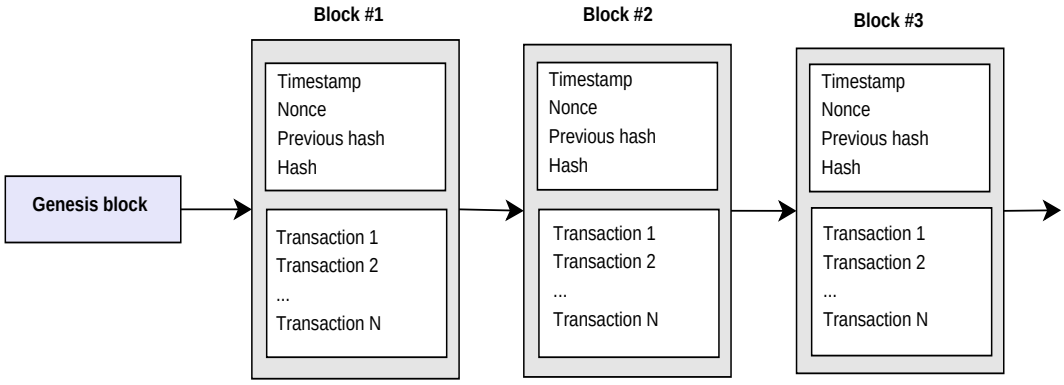


Figure 3.3: Structure of blockchain.

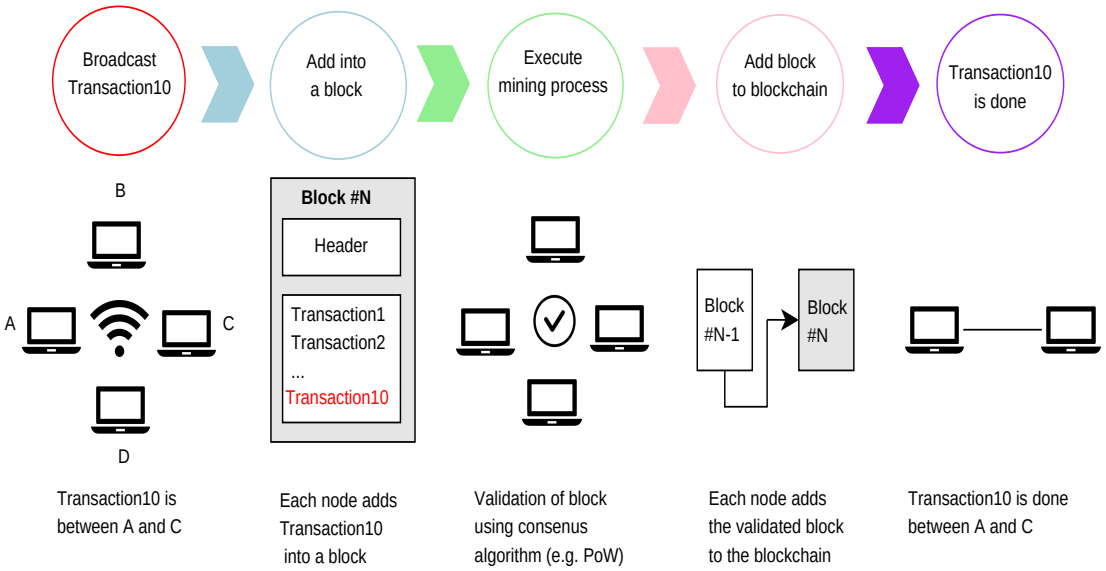


Figure 3.4: Validation of transaction.

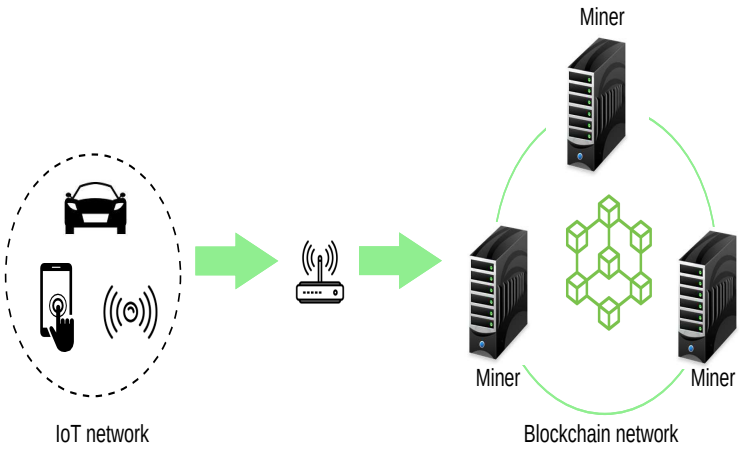


Figure 3.5: Blockchain architecture.

private one includes only defined nodes. Ethereum and Bitcoin are the most popular public blockchains. The selection of blockchain type and consensus algorithm depends on the nature and requirements of IoT application. Figure 3.5 demonstrates the architecture of blockchain in IoT.

Due to its prominent features such as decentralization, immutability, transparency, the blockchain technology can be applied in several IoT applications to provide authentication [103–107], access control [108–110] and trust management [111–115].

### 3.2.4 Lightweight cryptography-based solutions

Cryptography is an effective tool to guarantee confidentiality, integrity and authentication. However, most of IoT devices have challenging characteristics such as processing, memory and battery power. Thus, traditional cryptographic algorithms are not suitable for resource-constrained IoT devices. Recently, lightweight cryptographic primitives were proposed to secure IoT systems. As presented in Figure 3.6, lightweight cryptographic algorithms can be classified into four main classes: block ciphers, stream ciphers, hash functions and elliptic curve cryptography (ECC) [116].
In block ciphers, a block of plaintext is encrypted at a time, while stream ciphers encrypt/decrypt a single bit or byte of plaintext/ciphertext.
Hash functions are used to provide data integrity by generating a fixed-length message from an arbitrary-length message.
ECC is a lightweight asymmetric cryptographic technique that provides the same level of security as rivest-shamir-adleman (RSA) algorithm with smaller key size.

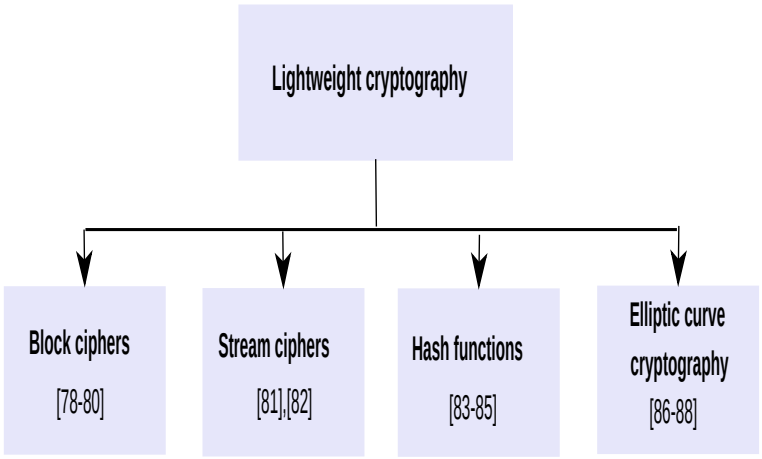


Figure 3.6: Lightweight cryptography for IoT.

The lightweight cryptographic techniques can be adopted to achieve key security requirements including confidentiality, integrity and authentication [117–121].

### 3.2.5 Homomorphic and searchable encryption-based solutions

The number of IoT devices is increasing to enable the creation of more intelligent applications. These devices generate a massive amount of data that needs to be gathered and analyzed. Cloud computing provides computation and storage services for IoT collected data. These data can be highly sensitive and thus need to be protected from unauthorized access. To provide privacy preservation, the collected data are encrypted then stored in the public cloud.

Homomorphic encryption (HE) allows calculations on encrypted data without revealing the original data. There are two basic types of homomorphic encryption: partially and fully homomorphic methods [122].

Searchable encryption (SE) enables secure search over encrypted data stored on a cloud server. The SE techniques include symmetric SE, asymmetric SE and attribute-based SE [123].

The proposed HE-based schemes and SE-based techniques aim to provide privacy-preserving in cloud-based IoT applications [124, 125].

### 3.2.6 Machine learning-based solutions

Machine learning (ML) is a promising technology that offers embedded intelligence to IoT devices to cope with different security issues. It is a subset of artificial intelli-

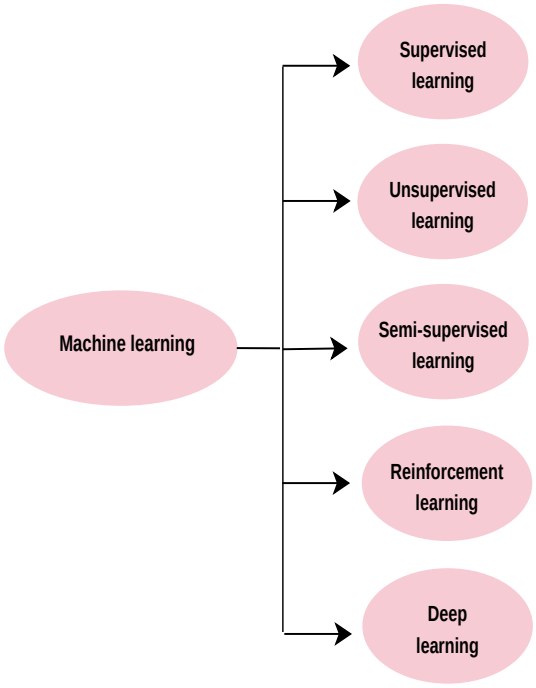


Figure 3.7: Machine learning algorithms.

gence (AI) that can be used to develop intelligent security systems for IoT networks. Various types of attack lunched on IoT systems such as DoS attack can be detected and mitigated using ML techniques. The ML algorithms can also be used to detect anomalies and intrusions in IoT networks [126, 127].

The ML algorithms are classified into five classes: supervised, unsupervised, semi-supervised, reinforcement and deep learning as shown in Figure 3.7.

Supervised learning algorithms such as support vector machines (SVM), decision tree (DT) and naive bayes (NB) are used to secure IoT systems. However, they require large storage and time for data training.

K-means clustering and hierarchical clustering are two common algorithms of unsupervised learning that do not require data training. The unsupervised algorithms are less efficient than supervised approaches.

Semi-supervised learning was introduced to reduce the datasets needed for training. Nevertheless, it does not provide detection accuracy compared to supervised learning.

Reinforcement learning techniques do not need rich training dataset, but require the knowledge of state transition function.

Deep learning techniques have been employed to address limitations of other ML techniques [126, 127]. Major deep learning algorithms such as convolutional neural network

(CNN), recurrent neural network (RNN), deep belief network (DBN), deep Q-netwrok (DQN) can be used to improve security in IoT systems.

## 3.3 Related work

Several research works were recently proposed to provide different security requirements for various IoT applications at the three layers of IoT architecture. A summary of the proposed security schemes for IoT is provided in Table 3.1.

Turkanovic et al. [128] proposed a lightweight authentication scheme for heterogeneous WSNs in the IoT. Their method is based on one-way hash function that requires lightweight computation and thus is energy-efficient. However, it is susceptible to various security weaknesses, as claimed in [129]. Later, Farash et al. [129] proposed an enhanced protocol to address the security flaws of Turkanovic et al.'s scheme.

To secure communication in IoT healthcare applications, Shen et al. [130] designed two authentication and key establishment protocols for wireless body area networks (WBANs). The two protocols are based on ECC and message authentication code (MAC) which provide confidentiality, integrity, and authenticity of data. However, they are vulnerable to various types of security attack, such as impersonation and replay attacks. Additionally, the two proposed schemes do not achieve mutual authentication, as the authors claimed.

Wu et al. [131] proposed an improved authentication scheme for WSNs-based IoT. The proposed scheme is based on ECC and one-way hash function and provides various IoT security properties. However, the mutual authentication phase is not efficient in terms of communication cost. Moreover, it is vulnerable to user impersonation and DoS attacks.

In [132] proposed a robust authentication protocol to secure multimedia communications in IoT-enabled WSNs. The proposed protocol meets several security requirements. However, it is not lightweight in terms of communication and storage costs.

The authors of [133] proposed an improved authentication and key agreement scheme based on ECC for WSNs. The proposed scheme achieves high level of security. However, it requires high computation and communication costs and therefore is not suitable for practical IoT applications.

As the IoT can be merged with the industry domain, Li et al. [134] proposed an ECC-based authentication protocol to secure WSNs in the IIoT environment. The proposed scheme is secure against various types of attack. However, it has a considerable communication overhead.

Mehmood et al. [135] proposed a secure mechanism called inter-cluster multiple key distribution scheme (ICMDS) for WSNs. They focused on authentication and key management in cluster-based WSNs. They employed different key distribution methods to secure inter-cluster communication in WSNs. However, the proposed scheme is insecure, because it is susceptible to several types of security attack.

Amin et al. [136] presented an efficient network architecture for WSNs and proposed a robust authentication protocol to secure data transmission. The proposed scheme provides mutual authentication and key agreement. However, it is inefficient in terms of communication cost since the network includes multiple base stations.

Gupta et al. [137] proposed a lightweight authentication and key agreement protocol based on one-way hash function and XOR operation. Their network consists of wearable devices, a mobile device or gateway, and an authentication server. The wearable devices must authenticate the gateway before sending health data with the assistance of the server. The proposed scheme provides data security. However, it is not efficient in terms of storage and communication costs.

Song et al. [138] proposed a privacy-preserving scheme to secure smart home systems. They outlined the state-of-art designs of smart home systems and presented a generic future smart home architecture. The collected data is encrypted using advanced encryption standard (AES) algorithm with 256-bits key. The data integrity is achieved using MAC. The proposed scheme secures the data transmission. However, it requires large resources for the AES S-box.

Aljawarneh et al. [139] proposed an encryption scheme to secure multimedia data in IoT. The symmetric proposed scheme uses lightweight genetic operations to encrypt the collected data, and thus provides data confidentiality. However, data integrity is not satisfied and the replay attack is not mitigated. Furthermore, the scheme has inefficient cost.

Elhoseny et al. [140] presented a secure data transmission approach for IoT healthcare systems. For medical data encryption, they combined RSA and AES algorithms

which are two landmarks in conventional cryptography. Their proposed hybrid scheme provides security and privacy of medical data. Nevertheless, it is not suitable for resource-constrained IoT devices.

Chaudhry et al. [141] proposed an improved remote user authentication scheme in IoT. The proposed scheme is based on ECC and cryptographic hash function. It uses password and smart card as two factors to provide mutual authentication and session key agreement. However, the login and authentication phase relies on a centralized server which makes the proposed scheme prone to single point of failure. Moreover, it cannot resist the user impersonation attack as the authors claimed.

Wazid et al. [142] presented a lightweight user authentication mechanism in the context of hierarchical IoT. The proposed scheme uses three factors; smart card, password and user biometrics and it is based on cryptographic hash function and symmetric cryptography. In this scheme, the user can access the information of IoT devices after authentication and session key establishment through a central controller. The proposed scheme provides user anonymity and intraceability. However, it is vulnerable to potential security attacks such as synchronization and DoS attacks. Furthermore, the data stored in IoT devices may not be accessible to the legitimate user on demand because of battery power exhaution.

Sharma et al. [143] designed a secure user authentication approach for cloud-based IoT applications. The remote user and the cloud server are mutually authenticated and share a session key to secure future communications. However, the proposed scheme is based on password and smart card and thus, it is prone to user's password leakage and stolen smart card attacks.

Lin et al. [144] designed an anonymous authentication scheme using blockchain technology and group signature. The proposed scheme enables users to remotely access smart home devices through a gateway node. To verify a transaction, the gateway node executes a smart contract and all valid transactions are added to the blockchain by consensus nodes. The proposed system does not achieve scalability because it is based on group signature where the number of users should be statically defined.

Deebak et al. [145] proposed a remote user authentication framework based on ECC and cryptographic hash function for smart healthcare IoT systems. The proposed scheme involves user's biometrics to resist the user impersonation attack. However,

it requires more computation overhead compared to non ECC-based authentication schemes.

Lee et al. [146] proposed an improved user authentication scheme for IoT. The proposed scheme is lightweight and suitable for constrained IoT environments. Howewer, the remote user directly authenticate and negociate a session key with the IoT device without involving a gateway node. As a result, energy conservation of IoT devices is not effective.

Cui et al. [147] presented a blockchain-based authentication mechanism to connect with sensor nodes in WSNs-enabled IoT. The remote user sends an authentication request to the blockchain deployed on base station of the WSN. The user is identified using its certificate distributed by a certificate authority (CA). However, the authors did not consider the certificate revocation or update which is highly required to mitigate certificate attacks.

Sadhukhan et al. [148] proposed an ECC-based user authentication and session key agreement scheme in IoT. The proposed scheme provides mutual authentication and session key agreement. However, it does not preserves user anonymity and intraceability. Moreover, it is inneficient in terms of computation overhead because it is based on cryptographic hash function, ECC and symmetric cryptography.

## 3.4 IoT security challenges

The number of IoT devices is soaring, and the amount of data is increasing as well. This growth is faced with several security issues which should be handled to ensure the evolution of the IoT into a secure infrastructure. Although the studied emerging solutions have been introduced to provide improved security in different IoT systems, they impose several security challenges that are not properly solved.

- Conventional security mechanisms cannot be fully integrated with IoT environments since the IoT devices have inherently limited resources. Therefore, the development of effective security solutions for tiny embedded devices is required. Unfortunately, some emerging technologies and approaches such as blockchain, homomorphic encryption, searchable encryption and machine learning algorithms require high processing and storage capabilities. Therefore, it is challenging to

Table 3.1: Summary of related work.

| **Related work** | **Security solution** |
|---|---|
| Turkanovic et al. [128] | Lightweight cryptography |
| Farash et al. [129] | Lightweight cryptography |
| Shen et al. [130] | Lightweight cryptography |
| Wu et al. [131] | Lightweight cryptography |
| Mishra et al. [132] | Lightweight cryptography |
| Wang et al. [133] | Lightweight cryptography |
| Li et al. [134] | Lightweight cryptography |
| Mehmood et al. [135] | Lightweight cryptography |
| Amin et al. [136] | Lightweight cryptography |
| Gupta et al. [137] | Lightweight cryptography |
| Song et al. [138] | Conventional cryptography |
| Aljawarneh et al. [139] | Conventional cryptography |
| Elhoseny et al. [140] | Conventional cryptography |
| Chaudhry et al. [141] | Lightweight cryptography |
| Wazid et al. [142] | Lightweight cryptography |
| Sharma et al. [143] | Lightweight cryptography |
| Lin et al. [144] | Blockchain |
| Deebak et al. [145] | Lightweight cryptography |
| Lee et al. [146] | Lightweight cryptography |
| Cui et al. [147] | Blockchain |
| Sadhukhan et al. [148] | Lightweight cryptography |

trade-off between security and performance in IoT infrastructure.

- In a dynamic, heterogeneous, and large-scale environment, adaptive trust models are required to enable devices to recognize trustworthy nodes. The IoT takes advantage of fog computing to achieve different security requirements. Fog nodes cooperate with each other to provide real-time and latency-sensitive services to IoT users. However, a fog node does not have any information about other nodes, it is challenging to ensure that all joining fog nodes are trusted. In fact, users have several fog nodes available to cooperate for guaranteeing IoT services. Thus, it is imperative to select trustworthy fog nodes.

- The IoT is rapidly spreading in different domains. Consequently, physical objects of daily life are progressively integrated in various environments and thus the scalability of systems needs to be ensured. However, the centralized SDN architecture cannot deal with a large number of IoT devices. In addition, the SDN-based solutions are not efficient in high dynamic IoT environments such as vehicular networks. Hence, it is necessary to enforce the scalability property in

Table 3.2: Security purposes and challenges of IoT security solutions.

| **Emerging solution** | **Security purpose** | **Security challenge** |
|---|---|---|
| Fog computing | Authentication, encryption | Trust management |
| SDN | Key management, identity management | Scalability |
| Blockchain | Authentication, access control, trust | Computation complexity, privacy |
| Lightweight cryptography | Confidentility, integrity, authentication | Key management |
| HE and SE | Privacy-preserving | Computation complexity |
| Machine learning | Anomaly detection, attack detection | Computation complexity |

SDN networks.

- As IoT devices are tremendously increasing, a massive amount of data including sensitive data are generated and exchanged via Internet. The blockchain technology efficiently tackles the scalability issue due to its distributed architecture. However, the blockchain does not ensure the privacy of transactions and it is prone to data leakage. In fog computing-based architecture, fog nodes are responsible for forwarding data to the cloud. If fog nodes are not trustworthy or compromised by an adversary, they can disclose personal information.
- The security of data transmission can be achieved using encryption techniques. The encryption of transmitted data prevents intruders from revealing the content of messages. This approach can be applied when the communication parties share encryption/decryption keys. In symmetric encryption (*i.e.,* block ciphers, stream ciphers and hash functions), the key must be pre-distributed or securely communicated. However, in scalable IoT environments, key management including distribution, agreement, update and revocation remains a meaningful task.

Table 3.2 summarizes the main security purposes and challenges of the studied emerging solutions.

## 3.5 Conclusion

Shortly, the IoT will be extended to Internet of everythings (IoE), the security of future IoT systems will be vital. Several research efforts are required to face the integration of IoT and emerging technologies to guarantee resilient and desirable level

of security. In this chapter, we presented several emerging solutions that promise to improve the security of IoT systems. We reviewed recent related work that were proposed to achieve IoT security. We also discussed the main security challenges that need to be addressed to propose new security schemes suitable for IoT environments.

In next chapter, we present our proposed methods that take into consideration the advantages and limitations of related work to enhance the security of IoT systems.

# Contributions

**Chapter 4:** Enhanced authentication and key management scheme for securing sata transmission in the Internet of Things

**Chapter 5:** Improved bio-inspired security scheme for privacy-preserving in the Internet of Things

**Chapter 6:** Lightweight blockchain-based remote user authentication for fog-enabled IoT deployment

# Chapter 4

# Enhanced authentication and key management scheme for securing data transmission in the Internet of Things

## 4.1 Introduction

The Internet of Things (IoT) is an emerging paradigm that has been recognized as a revolutionary technology of this century. It allows everyday objects to communicate seamlessly with each other to provide services without human intervention [149]. The ultimate goal of the IoT is to change human life gradually through its smartness and intelligence.

Wireless sensor networks (WSNs) constitute one of the enabling technologies of the IoT, because they are widely used in different IoT applications, such as environmental monitoring, disaster management, battlefield surveillance, industry, healthcare, and assisted living [70, 150–153].

A WSN typically consists of a large number of tiny sensors deployed deterministically by hand or randomly in the target environment. These sensors have limited resources because of their small size. They sense data from the environment and transmit them through wireless links. However, the sensed data can be deeply sensitive and intercepted by an unauthorized entity [154, 155].

To secure data transmission over public channel, a secret key should be shared between communicating parties to encrypt the transmitted data. However, it is highly required to verify the identity of the involved parties before negociating the cryptographic key. Mutual authentication and session key agreement are a must to ensure that only authorized entities can access the transmitted information [43, 156].

The proposed authentication and session key agreement schemes are based on ECC [130, 131, 133–135] and one-way hash function [128, 129, 132, 136, 137]. The ECC is more secure and suitable for constrained environments, because it provides high-level security with small key size.

In this chapter, we propose an enhanced scheme based on ECC to achieve authentication and session key agreement for WSNs in the context of IoT. The security of the enhanced scheme is formally verified using the Burrows-Abadi-Needham (BAN) logic and the Automated Validation of Internet Security Protocols and Applications (AVISPA) tool. A comparison to recent related methods [128–134, 136, 137] is also provided to show that our proposed scheme is more secure and efficient.

## 4.2 Preliminaries

### 4.2.1 Elliptic curve cryptography

ECC is a public key cryptography approach based on elliptic curves. An elliptic curve $E$ is the set of solutions defined by the cubic equation [157].

$$y^2 = x^3 + ax + b \quad \text{where} \quad 4a^3 + 27b^2 \neq 0 \tag{4.1}$$

In cryptography, elliptic curves are typically defined over a finite field of prime order. Let $p$ be a prime number, the integers [0, 1, ..., $p-1$] with addition and multiplication performed modulo $p$, is a finite field of order $p$ denoted by $F_p$ [157].

Point addition and point doubling are the basic elliptic curve operations. Point multiplication, referred to as scalar multiplication, is calculated using a series of addition and doubling [158] and defined as.

$$kP = P + P + P + ... + P(k \quad times) \tag{4.2}$$

Let $P$, $Q \in E(F_p)$ such that $Q = nP$, then, it is difficult to determine $n$ given $P$ and $Q$. This problem is called the elliptic curve discrete logarithm problem (ECDLP) [157]. The hardness of the ECDLP allows several cryptographic schemes based on elliptic curves.

### 4.2.2 Weil pairing

Weil pairing, which is denoted by $e$, takes as input a pair of points of order $m$ or m-torsion points and outputs an element of finite field that is the $m^{th}$ root of unity [159].

Let $G_1$ and $G_2$ be an additive and multiplicative group of order $m$, respectively. A bilinear pairing on ($G_1$, $G_2$) is a map $e$: $G_1 \times G_1 \rightarrow G_2$ that satisfies the following conditions:

— Bilinearity: for all $R$, $S$, $T \in G_1$ , $e(R+S,T) = e(R,T)e(S,T)$ and $e(R,S+T) = e(R,S)e(R,T)$.

— Non-degeneracy: $e(P,P) \neq 1$.

— Computability: $e$ can be efficiently computed.

If $P$, $Q \in G_1$, then $e(aP,bQ) = e(P,Q)^{ab}$. This property is additional means of defining bilinearity [160].

### 4.2.3 Network model

Our network architecture consists of a single BS and a large number (hundreds) of homogeneous and stationary sensors. The BS is assumed to be a powerful and reliable device. The sensors are deployed randomly or deterministically by hand in a target environment. To reduce the energy consumption of the sensors, the network is organized into clusters. Each cluster has a leader sensor node, referred as the cluster head (CH), which is selected using the approach presented in [161]. The CHs aggregate data sensed from their cluster's members (CMs) and send it to the BS directly or through another CH. If a CH receives a packet from another CH, it simply forwards it to the BS. Figure 4.1 shows the network architecture.

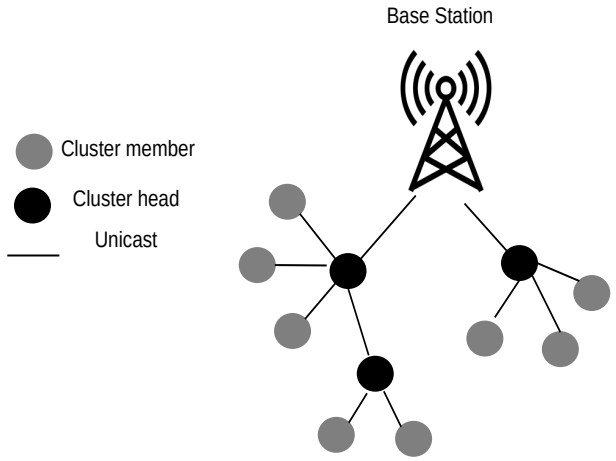


Figure 4.1: Network architecture of MAKA scheme.

## 4.3 Enhanced scheme

The name of our enhanced scheme, MAKA, stands for Mutual Authentication and Key Agreement. It consists of five phases: initialization, key generation, node registration, node authentication, and session key agreement. The notations used in our proposed scheme are listed in Table 5.1. Figure 4.2 shows a summary of the proposed scheme related to the key generation, node registration, node authentication, and session key agreement phases. The details of each phase are discussed in the following subsections.

### 4.3.1 Initialization

This phase is executed in offline mode. The BS generates the system parameters including an elliptic curve $E$ over a finite field of prime order $p$, an additive cyclic group $G$, a generator $g \in G$, a master key $k \in \mathbb{Z}_p$, and a hash function $H$: $(0,1)^* \rightarrow G$. Then, the BS generates its public key $Pu_{BS} = H(ID_{BS})$ and private key $Pr_{BS} = k * Pu_{BS}$. Each node is preloaded with a unique ID, the prime number $p$, the generator $g$, the master key $k$, the hash function $H$, the BS's identifier $ID_{BS}$, and the BS's public key $Pu_{BS}$.

### 4.3.2 Key generation

After deployment of the network, each node generates its public and private keys by computing $Pu_i = H(ID_i)$ and $Pr_i = k * Pu_i$, respectively. Then, it destroys the

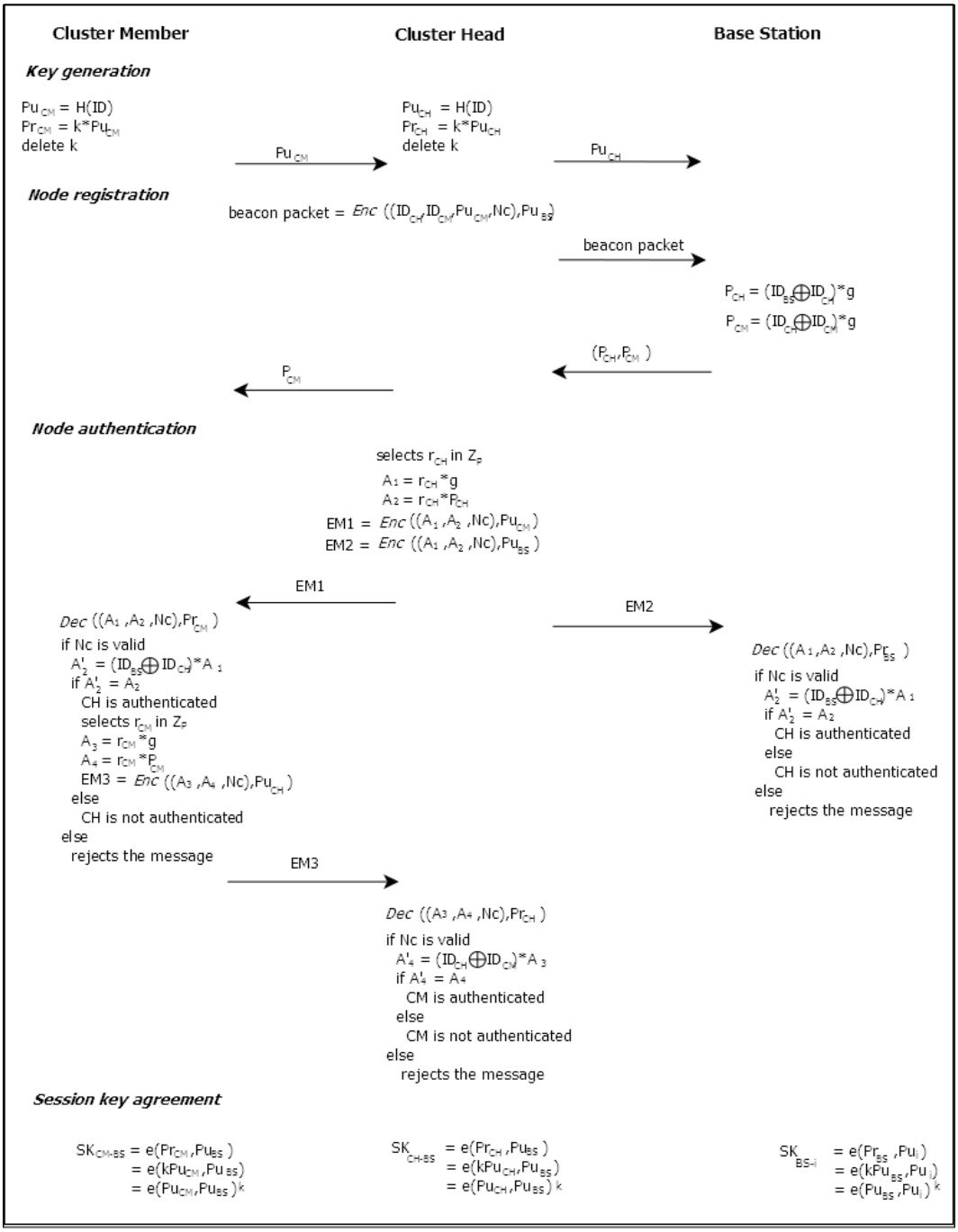


Figure 4.2: Summary of MAKA scheme.

Table 4.1: Notations used for MAKA scheme.

| **Notation** | **Description** |
|---|---|
| BS | Base station |
| CH | Cluster head |
| CM | Cluster member |
| $p$ | A large prime number |
| $\mathbb{Z}_p$ | Integer numbers less than p |
| $H$ | Hash function that maps string to elliptic point |
| $Pu_{BS}$ | Public key of BS |
| $Pr_{BS}$ | Private key of BS |
| $Pu_i$ | Public key of node i |
| $Pr_i$ | Private key of node i |
| $SK_{i-BS}$ | Session Key between node i and BS |
| $Nc$ | Nonce (a random number used once) |
| $e$ | Weil pairing |
| $\oplus$ | XOR operation |
| $*$ | Scalar multiplication |
| $Enc(m,k)$ | Asymmetric encryption of $m$ using the key $k$ |
| $Dec(m,k)$ | Asymmetric decryption of $m$ using the key $k$ |

master secret key $k$ and publishes its public key.

### 4.3.3 Node registration

When the CHs have been selected, they generate a beacon packet consisting of the CH's ID, CMs' IDs, CMs' public keys, and a nonce $Nc$. Then, each CH encrypts its generated beacon packet using the BS's public key and sends it to the BS.
The BS decrypts the received packet using its private key and verifies the nonce $Nc$. If the verification succeed (*i.e.,* $Nc$ is a fresh number), it generates a pseudo-identity $P_{CH} = (ID_{BS} \oplus ID_{CH}) * g$ for each CH and $P_{CM} = (ID_{CH} \oplus ID_{CM}) * g$ for each CM. Then, it sends securely these pseudo-identities to the respective CH.
The CH distributes securely the pseudo-identities among its CMs.

### 4.3.4 Node authentication

This phase is performed to authenticate each node before the data transmission process. In cluster-based sensor networks, it is more likely that the CHs will be attacked by intruders. Hence, they should be authenticated by their CMs and the BS.
Each CH selects a random number $r_{CH} \in \mathbb{Z}_p$, computes $A_1 = r_{CH} * g$ and $A_2 =$

$r_{CH} * P_{CH}$, and then sends Enc(($A_1$, $A_2$, $Nc$), $Pu_{CM}$) to its CMs and Enc(($A_1$, $A_2$, $Nc$), $Pu_{BS}$) to the BS.

**Mutual authentication between cluster head and cluster member**

The CM decrypts the received messages using its private key. Next, it verifies the received nonce; if it is non valid (*i.e.,* $Nc$ is not a fresh number), it rejects the message. Otherwise, it computes $A_2^{'} = (ID_{BS} \oplus ID_{CH}) * A_1$ and compares it to $A_2$. If $A_2^{'} = A_2$, the CH is authenticated.

Then, the CM selects a random number $r_{CM} \in \mathbb{Z}_p$, computes $A_3 = r_{CM} * g$ and $A_4 = r_{CH} * P_{CM}$, and sends Enc(($A_3$, $A_4$, $Nc$), $Pu_{CH}$) to the CH.

When the CH receives the message from the CM, it decrypts it using its private key and verifies $Nc$. If it is non valid, it rejects the message. Otherwise, it computes $A_4^{'} = (ID_{CH} \oplus ID_{CM}) * A3$ and compares it to $A_4$. If $A_4^{'} = A_4$, the CM is authenticated.

**Authentication of cluster head at base station**

The BS decrypts the message received from the CH using its private key Dec(($A_1$, $A_2$, $Nc$), $Pr_{BS}$). Then, it verifies $Nc$; if it is non valid, it rejects the message. Otherwise, it computes $A_2^{'} = (ID_{BS} \oplus ID_{CH}) * A_1$ and compares it to $A_2$. If $A_2^{'} = A_2$, the CH is authenticated.

### 4.3.5 Session key agreement

After the authentication phase, a session key is generated to secure communication in the network. In ICMDS, the authors focused on securing inter-cluster communication. However, in our study we aimed to secure all communications (*i.e.,* intra and inter-cluster communications).

Each sensor $node_i$ computes $SK_{i-BS} = e(Pr_i, Pu_{BS}) = e(kPu_i, Pu_{BS}) = e(Pu_i, Pu_{BS})^k$

The BS computes $SK_{BS-i} = e(Pu_i, Pr_{BS}) = e(Pu_i, kPu_{BS}) = e(Pu_i, Pu_{BS})^k$ and stores all shared keys on a database.

In our proposed scheme, the BS shares a secret session key $e(Pu_i, Pu_{BS})^k$ with each $node_i$ in the network. These shared keys are used for data encryption and decryption. Before data transmission, each cluster member encrypts the sensed data using $SK_{CM-BS}$ and sends the data to its CH. The CH encrypts the aggregated received data using the shared key $SK_{CH-BS}$ and sends the encrypted data to the BS. To reduce the

energy consumption of the sensor nodes, the decryption of sensed data is performed only by the BS, which is assumed to be a powerful entity.

## 4.4 Security evaluation

### 4.4.1 Informal security analysis

**Replay attack**

In the proposed scheme, an attacker may intercept the messages transmitted between the CM and CH or between the CH and BS during the node registration and authentication phases. Then, the attacker can attempt to launch a replay attack by retransmitting these messages. However, in our scheme all communication messages include an encrypted nonce, and thus, the attacker is unable to modify them. If the message is replayed, the receiver detects the replayed message by verifying the nonce. Hence, our proposed scheme is secure against replay attacks.

**Denial of Service (DoS) attack**

In our scheme, the CH aggregates only the data received from authenticated nodes to forward them to the BS. The latter accepts only data received from an authenticated CH. Furthermore, the CH or BS rejects replayed data. As a result, the MAKA scheme resists DoS attacks.

**Cluster head impersonation attack**

As discussed above, an attacker cannot replay the messages transmitted by CHs. Moreover, even if the attacker claims to be a CH, each CM should authenticate the CH before sending the sensed data. Because all the nodes are preloaded with the generator $g$, the attacker cannot compute $A_1$; thus, it cannot be authenticated. Therefore, the proposed scheme resists a CH impersonation attack.

**Mutual authentication**

The proposed scheme MAKA ensures mutual authentication between CMs and the CH. In fact, pseudo-identities are sent securely in the network. An adversary cannot be authenticated as a legitimate node (*i.e.,* a CM or a CH), because he/she cannot compute $A_2$ or $A_4$ without having the pseudo-identities.

**Sybil attack**

An adversary can assume the identity of a sensor node to be authenticated as a le-

gitimate node. However, in our scheme sensor nodes are preloaded with a unique ID and the attacker cannot determine the identity from messages transmitted within the network. Furthermore, every node performs the authentication phase before sending or receiving data. The adversary cannot be authenticated unless he/she has the preloaded parameters. Thus, the proposed scheme is resilient to Sybil attacks.

**Session key secrecy**

Every node in the network shares a symmetric session key with the BS. The session key is computed mutually by the node and BS for securing data transmission. An attacker cannot generate a valid session key, because he/she has no knowledge of the parameters used for the asymmetric key generation phase. However, even if an attacker compromises a node and acquires access to the session key, it does not affect the security of the system.

**Eavesdropping attack**

The proposed scheme achieves mutual authentication, which prevents the attacker from acquiring access to information flowing in the network during the transmission process. In addition, all sensed data are encrypted using symmetric session keys. This mechanism provides data confidentiality and prevents the intruder from divulging the information.

### 4.4.2 Formal security proof using BAN logic

BAN logic is widely used for the authentication protocols to prove the mutual authentication property and establishment of a secure session key [162]. To analyze a protocol using the BAN logic, we derive the idealized form of the protocol from the original form, and give the initial assumptions (*i.e.,* shared keys, generated nonces, trusted parties or principals). Then, we apply the BAN logic rules (postulates) until the goals of the protocols are achieved. The BAN logic notations and rules are detailed in [162].

The goals that our proposed scheme should achieve are:

— $Goal1 : BS \mid\equiv BS \overset{SK}{\longleftrightarrow} CH$

— $Goal2 : BS \mid\equiv CH \mid\equiv BS \overset{SK}{\longleftrightarrow} CH$

— $Goal3 : CM \mid\equiv CH \mid\equiv (A_1, A_2, Nc_{CH})$

— $Goal4 : CH \mid\equiv CM \mid\equiv (A_3, A_4, Nc_{CM})$

The idealized form of our proposed scheme is given as follows:

— $M_1 : CH \rightarrow CM : \{A_1, A_2, Nc_{CH}\}_{Pu_{CM}}$

— $M_2 : CH \rightarrow BS : \{A_1, A_2, Nc_{CH}\}_{Pu_{BS}}$

— $M_3 : CM \rightarrow CH : \{A_3, A_4, Nc_{CM}\}_{Pu_{CH}}$

The initial assumptions of our proposed scheme are listed as follow:

— $AS_1 : BS \mid\equiv \stackrel{Pu_{BS}}{\longmapsto} CH$

— $AS_2 : CM \mid\equiv \stackrel{Pu_{CM}}{\longmapsto} CH$

— $AS_3 : CH \mid\equiv \stackrel{Pu_{CH}}{\longmapsto} CM$

— $AS_4 : BS \mid\equiv \#Nc_{CH}$

— $AS_5 : CM \mid\equiv \#Nc_{CH}$

— $AS_6 : CH \mid\equiv \#Nc_{CM}$

Using *seeing rule* on $M_1$, we get

$R_1 : CM \triangleleft \{A_1, A_2, Nc_{CH}\}_{Pr_{CM}}$

Using *message meaning rule* on $R_1$ and $AS_2$, we get

$R_2 : CM \mid\equiv CH \mid\sim (A_1, A_2, Nc_{CH})$

Using *nonce verification rule* on $R_2$ and $AS_5$, we get

$R_3 : CM \mid\equiv CH \mid\equiv (A_1, A_2, Nc_{CH})$ (Goal3 achieved)

Using *seeing rule* on $M_2$, we get

$R_4 : BS \triangleleft \{A_1, A_2, Nc_{CH}\}_{Pr_{BS}}$

Using *message meaning rule* on $R_4$ and $AS_1$, we get

$R_5 : BS \mid\equiv CH \mid\sim (A_1, A_2, Nc_{CH})$

Using *nonce verification rule* on $R_5$ and $AS_4$, we get

$R_6 : BS \mid\equiv CH \mid\equiv (A_1, A_2, Nc_{CH})$

Using *belief rule* on $R_6$, we get

$R_7 : BS \mid\equiv CH \mid\equiv (Nc_{CH})$

Using *session key rule* on $R_7$ and $AS_4$, we get

$R_8 : BS \mid\equiv BS \stackrel{SK}{\longleftrightarrow} CH$ (Goal1 achieved)

Using *nonce verification rule* on $R_8$ and $AS_4$, we get

$R_9 : BS \mid\equiv CH \mid\equiv BS \stackrel{SK}{\longleftrightarrow} CH$ (Goal2 achieved)

Using *seeing rule* on $M_3$, we get

$R_{10} : CH \triangleleft \{A_3, A_4, Nc_{CM}\}_{Pr_{CH}}$

Using *message meaning rule* on $R_{10}$ and $AS_3$, we get

$R_{11} : CH \mid\equiv CM \mid\sim (A_3, A_4, Nc_{CM})$

Using *nonce verification rule* on $R_{11}$ and $AS_6$, we get

$R_{12} : CH \mid\equiv CM \mid\equiv (A_3, A_4, Nc_{CM})$ (Goal4 achieved)

### 4.4.3 Formal security verification using AVISPA

AVISPA is the most widely used tool for the automated validation of Internet security protocols and applications. It provides a formal language called the High Level Protocol Specification Language (HLPSL) for specifying the intended security properties of the protocol. HLPSL is a role-based language that defines basic roles, composed roles or session, goal (*i.e.,* the intended security properties), and the environment role or top-level role [163, 164].

AVISPA also integrates four back-ends: On-the-fly Model-Checker (OFMC), Constraint Logic-based Attack Searcher (CL-AtSe), SAT-based Model-Checker (SATMC), and Tree Automata based on automatic approximations for the analysis of security protocols (TA4SP), which analyze and verify the security protocols by implementing various automatic analysis techniques [164].

The intended security properties need to be specified in HLPSL. The HLPSL is converted automatically to intermediate format (IF) using the HLPSL2IF translator. The IF is analyzed by a back-end specified by the user. The back-end verifies that the security goals are satisfied and produces output format (OF) [165].

To verify the proposed scheme formally using the AVISPA tool, we implemented three basic roles for the CM, CH, and BS in HLPSL related to the node registration phase and node authentication phase. Moreover, we specified the roles for session, goal, and environment. The session role instantiates the basic roles with concrete arguments and thus, describes one session of the protocol. The top-level role (environment) contains global constants and a composition of sessions, where the intruder participates in the execution as a legitimate node. The goals of our specification are the secrecy of identities and the generator, and the authentication of nodes.

We simulated our proposed scheme using the SPAN (Security Protocol ANimator for AVISPA) simulator and verified our proposed scheme under the widely used OFMC back-end. The verification results shown in Figure 6.4 indicate that the security goals

```
% OFMC
% Version of 2006/02/13
SUMMARY
 SAFE
DETAILS
 BOUNDED_NUMBER_OF_SESSIONS
PROTOCOL
 /home/span/span/testsuite/results/maka.if
GOAL
 as_specified
BACKEND
 OFMC
COMMENTS
STATISTICS
 parseTime: 0.00s
 searchTime: 8.15s
 visitedNodes: 1501 nodes
 depth: 7 plies
```

Figure 4.3: Formal verification results of MAKA scheme.

of the proposed scheme are satisfied. Therefore, our proposed protocol is secure against replay and eavesdropping attacks.

## 4.5 Performance analysis

### 4.5.1 Computation cost

We focus only on computation overhead on constrained sensor nodes, because the BS is considered a powerful device. We define the following notations.

— $T_{HG}$ is the cost of hash function on G.

— $T_{SM}$ is the cost of ECC scalar multiplication.

— $T_{E/D}$ is the cost of asymmetric encryption/decryption.

— $T_P$ is the cost of pairing on G.

We ignore the computational cost of XOR operation, because it is an inexpensive operation.

Table 6.3 presents the computation overhead of the CM and the CH during the various phases of the proposed scheme. Both the CM and CH generate the asymmetric key during the key generation phase, which takes $1T_{HG}+1T_{SM}$. In the node registration phase, the CH encrypts the beacon packet using the BS's public key, which requires

Table 4.2: Computational cost of MAKA scheme.

| **MAKA phases** | **CM side** | **CH side** |
|---|---|---|
| Key generation | $1T_{HG}$+$1T_{SM}$ | $1T_{HG}$+$1T_{SM}$ |
| Node registration | - | $1T_{E/D}$ |
| Node authentication | $3T_{SM}$+$2T_{E/D}$ | $3T_{SM}$+$3T_{E/D}$ |
| Session key agreement | $1T_P$ | $1T_P$ |
| **Total cost** | $1T_{HG}$+$4T_{SM}$+$2T_{E/D}$+$1T_P$ | $1T_{HG}$+$4T_{SM}$+$4T_{E/D}$+$1T_P$ |

$1T_{E/D}$. During the node authentication phase, the CM computes $A_1$ and $A_2$, and then encrypts the authentication request twice. The CM must decrypt and verify the CH authentication request. Then, the CM computes $A_3$ and $A_4$ and encrypts its authentication request. The CH must decrypt and verify the CM authentication request. Thus, the time required for the CM is $3T_{SM}$+$2T_{E/D}$, for the CH is $3T_{SM}$+$3T_{E/D}$.

### 4.5.2 Communication cost

To calculate the communication overhead of our proposed scheme, we defined the following assumptions.

— Identity and cryptographic nonce are of 160 bits.

— Elliptic points are of 320 bits.

— Encrypted data size is equal to plain data size.

Table 6.4 presents the communication overhead of the MAKA scheme between the CM, CH, and BS. During the key generation phase, the CM and the CH publish their public keys, each of which has 320 bits. During the node registration phase, the CH sends the beacon packet, which requires 800 bits, to the BS. The latter transmits the pseudo-identities $P_{CH}$ and $P_{CM}$, which require 640 bits. The CH forwards $P_{CM}$, which consists of 320 bits, to the CM. During the node authentication phase, three encrypted messages are transmitted between the CM, CH, and BS. The size of the encrypted message is 800 bits. There is no communication during the session key agreement process.

Table 4.3: Communication cost of MAKA scheme.

| **MAKA phases** | **No. of messages** | **No. of bits** |
|---|---|---|
| Key generation | 2 | 320+320=640 |
| Node registration | 3 | 800+640+320=1760 |
| Node authentication | 3 | 800+800+800=2400 |
| Session key agreement | - | - |

Table 4.4: Storage cost of MAKA scheme.

| **MAKA phases** | **Storage cost(bits)** |
|---|---|
| After key generation | 1440 |
| After node registration | 2240 |
| After node authentication | 640 |
| After session key agreement | 160 |

### 4.5.3 Storage cost

We evaluate the storage cost of the sensor node because it has tiny and limited memory. Table 6.5 presents the length of variables that the sensor node has to save in its memory after the different phases of our proposed scheme. It can be noted that a sensor node can delete or add some values into its memory while executing the MAKA scheme. From Table 6.5, it can be observed that the sensor node has to store 1440 bits, at the end of the key generation phase. After the node registration phase, the sensor node stores 2240 bits, into its memory. While at the end of the node authetication and session key agreement phases, the memory contains 640 bits, and then it is reduced to 160 bits.

## 4.6 Comparative analysis

In this section, we provide a comprehensive comparison of our proposed scheme with Mehmood et al.'s scheme [135] and related methods [128–134, 136, 137] in terms of security and efficiency.

The network architecture in [131–134, 137] consists of a user, BS/gateway, and sensor nodes, which is slightly different from our network model, which includes BS, CHs, and CMs. Hence, we consider the overhead between the BS and sensor nodes. We expect the computation cost of our proposed scheme to be greater than that of

Table 4.5: Computation cost comparison of MAKA scheme.

| **Scheme** | **Computation cost** | **Computation time** |
|---|---|---|
| MAKA | $6T_{SM}$+$5T_{E/D}$+$3T_P$ | 50.039 |
| Mehmood et al. [135] | $3T_{HG}$+$7T_{SM}$+$2T_{PA}$+$2T_P$ | 64.5156 |
| Turkanovic et al. [128] | $12T_H$ | 0.0276 |
| Farash et al. [129] | $22T_H$ | 0.0506 |
| Shen et al. [130] | $2T_{HG}$+$21T_{SM}$+$11T_{PA}$+$2T_S$+$4T_H$+$6T_{MAC}$ | 71.9448 |
| Wu et al. [131] | $17T_H$+$2T_{SM}$ | 0.0967 |
| Mishra et al. [132] | $18T_H$ | 0.0414 |
| Wang et al. [133] | $19T_H$+$2T_{SM}$ | 0.1013 |
| Li et al. [134] | $10T_H$+$2T_{SM}$ | 0.0806 |
| Amin et al. [136] | $26T_H$ | 0.0598 |
| Gupta et al. [137] | $10T_H$ | 0.023 |

other schemes.

To compare the computation cost, we define the time consumed by the hash function on $G$ $T_{HG} = 12.419$ ms, ECC scalar multiplication $T_{SM} = 2.226$ ms, ECC point addition $T_{PA} = 0.0288$ ms, asymmetric encryption/decryption $T_{E/D} = 3.85$ ms, pairing on $G$ $T_P = 5.811$ ms, symmetric encryption/decryption $T_S = 0.0046$ ms, one-way hash function $T_H = 0.0023$ ms, and MAC $T_{MAC} = 0.0046$ ms [166].

Table 4.5 summarizes the computation cost related to the authentication and key agreement phases in our proposed scheme and the schemes presented in [128–137]. As compared to those in [130, 135], our scheme is clearly more efficient, because it requires less computation. In contrast, it costs more than other schemes, because it is based on ECC, whereas the schemes in [128, 129, 131–134, 136, 137] are based on one-way hash function. The hash function requires lightweight computations as compared to ECC. However, ECC is more resilient to security attacks.

In WSNs, the BS is a powerful device and has more computational capabilities than the CHs and CMs/sensor nodes. Hence, it is essential to develop energy-efficient schemes for WSNs to enhance the lifetime of the network. According to [167], the energy consumption of sensor nodes depends on the length of the transmitted and received messages and the distance of the node from the destination.

To compare the communication cost, we make the same assumptions as defined previously. We also assume that the timestamp, distance, shared key, passwords, random number, and output of one-way hash function consist of 160 bits.

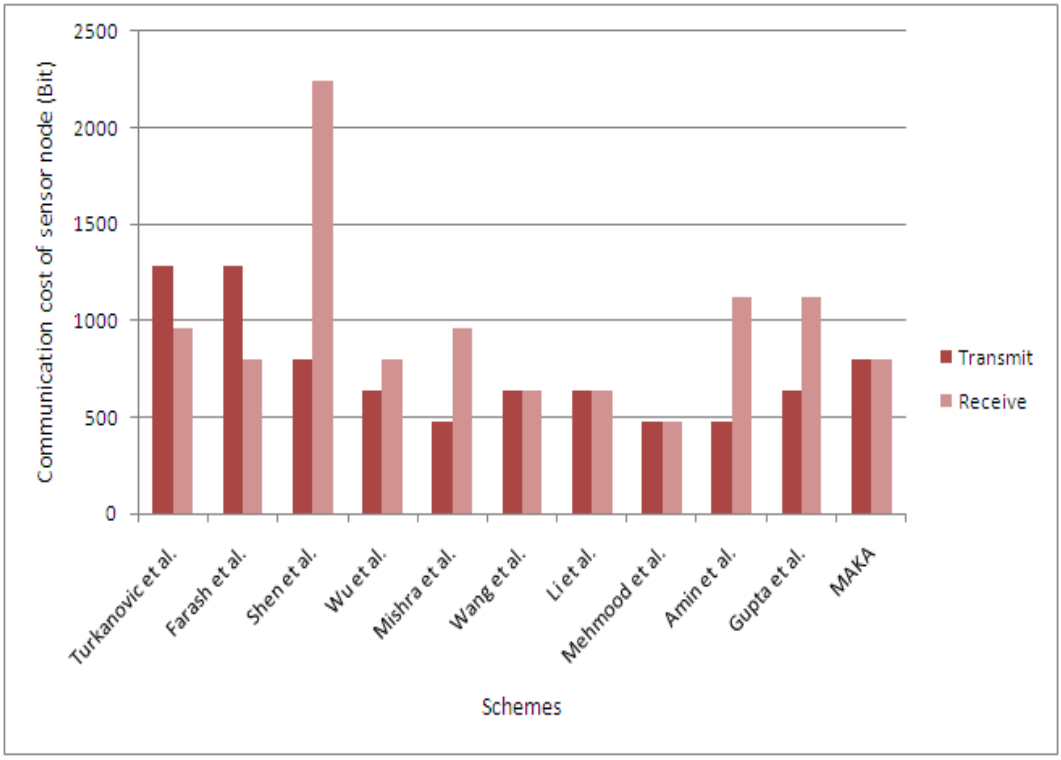


Figure 4.4: Communication cost comparison of MAKA scheme.

Figure 4.4 shows the communication overhead of a sensor node related to the authentication and key agreement phases. It is observed that our scheme has a small increased communication cost as compared to that of Mehmood et al.'s scheme. The additional cost provides mutual authentication and session key agreement, which is not achieved by Mehmood et al.'s scheme. In addition, our scheme is more efficient than those presented in [128–130] and thus requires less energy for the sensor nodes. As compared to the protocols presented in [131–134, 136, 137], our scheme has a small increase in the number of messages. However, in [131–134, 137] a sensor node communicates directly with the BS, which requires a large amount of power, even if the length of the transmitted and received messages is small. In our proposed scheme, the distance between a sensor node and the destination is smaller than the schemes presented in [131–134, 137]. As a result, the sensor nodes in our proposed scheme consume less energy than in other related schemes [131–134, 137].

Figure 4.5 presents the storage cost of a sensor node related to the authentication and session key agreement phases. For an objective analysis, the length of each variables are determined previously. It can be noted that a sensor node in our scheme has to store 640 bits, in its memory after executing the authentication phase, whereas in

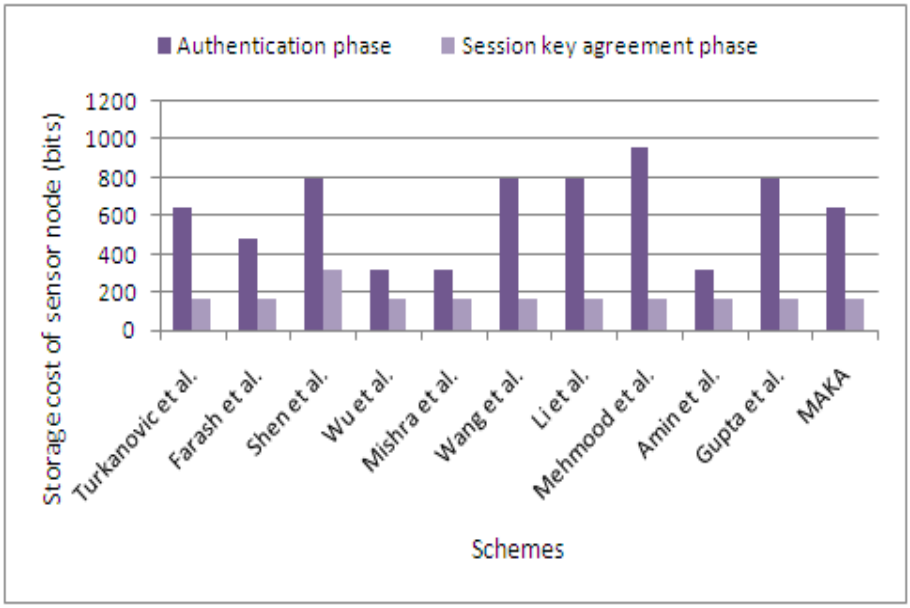


Figure 4.5: Storage cost comparison of MAKA scheme.

Mehmood et al.'s scheme [135], it has to store 960 bits. Our proposed scheme is more efficient than the schemes presented in [130, 133, 135, 137]. However, it requires more storage cost compared to [129, 131, 132, 136], but it is practically insignificant, because a typical sensor node (*e.g.,* Crossbow MICA2) has 128KB of memory space [128, 129].

Table 4.6 presents a comparison of the security features of the proposed and other schemes [128–137] . We can see that our proposed scheme is robust to all the mentioned security weaknesses of Mehmood et al.'s scheme [135]. The protocols presented in [128, 129, 132, 133, 136, 137] are susceptible to the clock synchronization problem, because they use timestamps. However, our proposed scheme resists known security attacks, provides data confidentiality and freshness, and avoids the clock synchronization problem. Hence, it is robust and suitable for different IoT applications.

If we integrate both the security and the performance results, we can conclude that our proposed scheme overcomes the security weaknesses of Mehmood et al.'s scheme. Moreover, it is more secure than other related methods. In terms of efficiency (*i.e.,* computation, communication and storage costs), the proposed scheme outperforms the schemes in [130,135]. In terms of energy consumption, the sensor nodes in our proposed scheme consume less energy than other related methods [131–134,137], as transmission energy is greater than computation energy.

Table 4.6: Comparison of security features of MAKA scheme.

| | MAKA | [135] | [128] | [129] | [130] | [131] | [132] | [133] | [134] | [136] | [137] |
|---|---|---|---|---|---|---|---|---|---|---|---|
| F1 | ✓ | × | ✓ | ✓ | × | × | ✓ | ✓ | × | ✓ | ✓ |
| F2 | ✓ | × | ✓ | ✓ | × | × | ✓ | ✓ | × | ✓ | ✓ |
| F3 | ✓ | × | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| F4 | ✓ | × | × | × | × | × | ✓ | ✓ | ✓ | × | ✓ |
| F5 | ✓ | × | × | × | × | ✓ | ✓ | ✓ | ✓ | × | ✓ |
| F6 | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| F7 | ✓ | × | × | × | × | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| F8 | ✓ | × | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| F9 | ✓ | ✓ | × | × | ✓ | ✓ | × | × | ✓ | × | × |

F1: Replay attack resistance, F2: DoS attack resistance, F3: Eavesdopping attack resistance, F4: Impersonation attack resistance, F5: Device Anonymity, F6: Session key secrecy, F7: Mutual authentication, F8: Session key agreement, F9: No clock synchronization

## 4.7 Conclusion

In this chapter, we proposed a mutual authentication and session key agreement (MAKA) scheme to secure communications in IoT-enabled WSNs. The informal security analysis shows that our proposed scheme is resilient to known security attacks. Moreover, we formally validated the MAKA scheme using the BAN logic and the widely used AVISPA tool. In comparison with Mehmood et al.'s scheme and recently developed related methods, both the security and performance results show that our proposed scheme is more secure and efficient for WSN-based IoT applications.

In next chapter, we extend our work to secure sensed data transmission using the generated session key. The collected data is encrypted using a lightweight symmetric technique before being transmitted to a public cloud server.

# Chapter 5

# Improved bio-inspired security scheme for privacy-preserving in the Internet of Things

## 5.1 Introduction

The IoT is a technology of the new era that aims to enhance people's quality of life by connecting physical world to digital world. It enables everyday objects to seamlessly communicate with each other through the internet to offer daily services without human intervention [168].

Different existing technologies such as wireless sensor networks (WSNs), radio frequency identification (RFID) and cloud computing are involved in the deployment of the IoT [43]. WSN technology plays a major part in the IoT since it provides sensing services to real-world objects and can be applied in a wide range of IoT applications [150]. Currently, sensors are designed to collect different types of information (*e.g.,* scalar data, images, audios and videos) from the physical world in real-time. This resulted in the emergence of wireless multimedia sensor networks (WMSNs) [169].

The IoT development provides several smart applications in healthcare, transportation, industry, business and so on [170]. In such applications, the IoT devices can collect and share sensitive data through wireless channels which allows attackers to disclose private information. Therefore, it is mandatory to guarantee privacy-preserving in IoT environments [171].

Cryptography is an effective method for protecting transmission of data in wireless channels [172]. It includes encryption and decryption processes and has two main types: symmetric and asymmetric techniques. One cryptographic key is used to encrypt and decrypt the data in symmetric cryptography while asymmetric cryptography involves two keys, a public key for encryption and a private key for decryption [155]. Conventional cryptographic algorithms are not suitable for resource-constrained IoT devices because they require large resources (*i.e.,* processing and memory). Consequently, achieving a good level of security with lightweight operations is challenging [173].

Recently, lightweight cryptography has significantly drawn attention. It aims to optimize conventional cryptographic algorithms and provide lightweight security solutions for resource-constrained devices [174]. As a sub-field of lightweight cryptography, bio-inspired cryptography intends to solve optimization in terms of security based on biological mechanisms [175].

In this chapter, we propose an improved symmetric bio-inspired security scheme to achieve privacy-preserving in the IoT. The proposed scheme is based on a genetic algorithm (GA) and a chaotic system to secure multimedia communications in the IoT environments. Then, we formally verified the security of the proposed scheme using AVISPA tool and evaluate its performance using NS-3. We also provide a comparison in term of security requirements and performance costs to recent related methods [138–140].

## 5.2 Preliminaries

### 5.2.1 Genetic algorithm

GA is an optimization algorithm based on Darwin's theory of evolution that carries out genetic operators including selection, crossover and mutation to generate new population. In selection, chromosomes are chosen from the population based on a fitness value. In crossover, two chromosomes produce a new chromosome based on a crossover point. In mutation, one bit in each chromosome is flipped. The crossover and mutation points are generated randomly [176]. Figure 5.1 and Figure 5.2 illustrate the crossover and mutation operations, respectively.

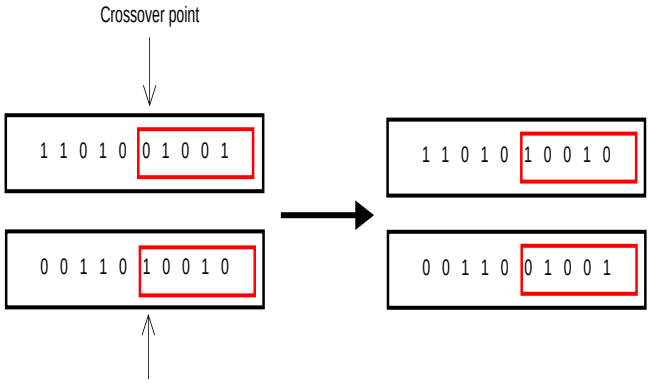


Figure 5.1: Crossover function.

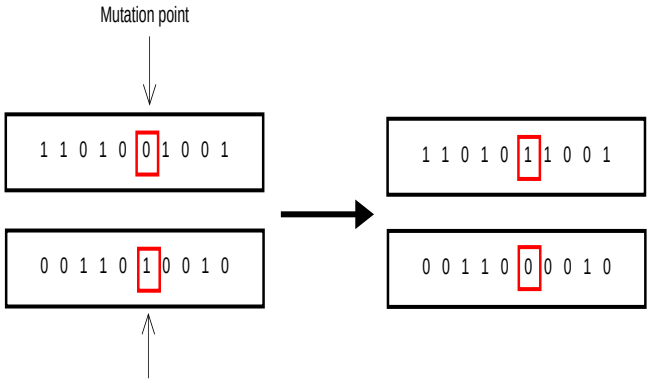


Figure 5.2: Mutation function.

### 5.2.2 Chaos theory

Chaos-based cryptography has been widely adopted for securing communications. A chaotic system is extremely sensitive to the initial conditions, which means that the outputs of two chaotic systems with slightly different inputs can be signicantly different.

Logistic map is a simple and nonlinear chaotic function that produces a series of numbers with randomness properties [177]. Equation 6.3 presents the chaotic logistic map.

$$\phi_{n+1} = \lambda\phi_n(1 - \phi_n) \tag{5.1}$$

where $\lambda$, known as bifurcation parameter, is a positive constant $0 < \lambda < 4$ and $0 <= \phi_0 <= 1$ is a start seed.

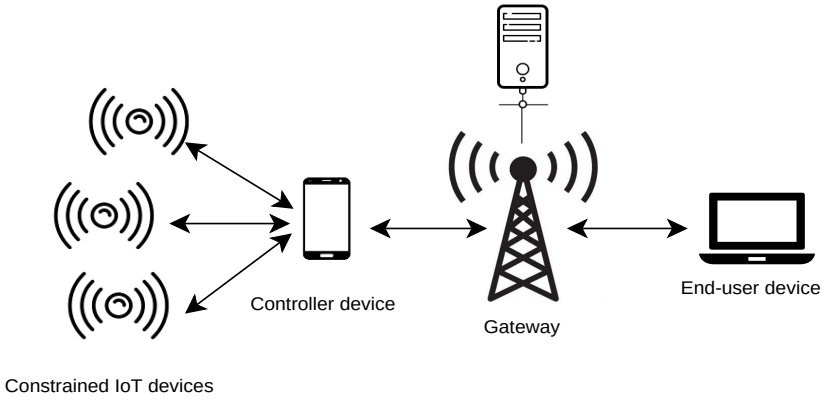


Figure 5.3: Network architecture of BOSS scheme.

### 5.2.3 Hash-based message authentication code

HMAC is a mechanism that integrates a cryptographic hash function (*e.g.,* MD5, SHA-1, SHA-2,...) with a secret shared key to provide data integrity and authenticity [178]. The receiver computes the HMAC on the received data and compares the result with the received HMAC. If the two values match, then the data has not been altered.

### 5.2.4 Network model

Our network model has four main components: constrained IoT devices, a controller device, a smart gateway and an end-user device. Figure 5.3 presents the network architecture of the proposed scheme.

— *Constrained IoT device*: is a multimedia sensor node that monitors a physical environment and encrypts the sensed data, then sends it to the controller device periodically. It has limited resources (*i.e.,* battery power, memory and processing).

— *Controller device*: represents the head of a cluster of constrained IoT devices with significant computational capabilities. It is responsible for collecting the encrypted data and forwarding it to the gateway node.

— *Smart gateway*: acts as an intermediary node between controller device and end-user device. It integrates a local server that initializes the system, registers new IoT devices and decrypts the received data. The gateway is assumed to be secure and trustworthy.

— *End-user device*: can access the gateway server and read decrypted data in order to take relevant decisions.

### 5.2.5 Energy model

In our network architecture, the constrained IoT devices are considered as sensor nodes powered with small batteries. In such environment, the energy conservation is a crucial task to prolonge the network lifetime. Equation 5.2 and Equation 5.3 are used to calculate the energy consumption of transmission and reception of $l$-bits message over a distance $d$, respectively [167].

$$E_T(l,d) = \begin{cases} lE_{elec} + l\epsilon_{fs}d^2 & if d < d_0 \\ lE_{elec} + l\epsilon_{mp}d^4 & if d \geq d_0 \end{cases} \tag{5.2}$$

$$E_R(l) = lE_{elec} \tag{5.3}$$

where $E_{elec}$ is the energy required by the electronic circuit, $\epsilon_{fs}$ and $\epsilon_{mp}$ are the energy required by the amplifier in free space and multi-path model, respectively, and $d_0$ is usually calculated using Equation 5.4.

$$d_0 = \sqrt{\frac{\epsilon_{fs}}{\epsilon_{mp}}} \tag{5.4}$$

### 5.2.6 Threat model

We consider the Dolev-Yao attack model [179] where the adversary $\mathcal{A}$ is assumed to perform the following types of attacks in the network.

— $\mathcal{A}$ can get private information by secretly overhearing the transmission of data over public channels.

— $\mathcal{A}$ can inject false data, alter or delete important information in the transmitted message.

— $\mathcal{A}$ can intercept the transmitted data and resend it maliciously to misdirect the receiver or suspend its services.

— $\mathcal{A}$ cannot perform physical attacks to retrieve information stored in IoT devices.

Table 5.1: Notations used for BOSS scheme.

| **Notation** | **Description** |
|---|---|
| $plaindata$ | Sensed data |
| $cipherdata$ | Encrypted data |
| $cipherkey$ | Encrypted key |
| $Crossover$ | Crossover function |
| $Mutation$ | Mutation function |
| $SK_i$ | Master key shared between sensor $i$ and gateway |
| $k_i$ | Sub-key of sensor $i$ |
| $x_0$ | Start seed of the first logistic map |
| $y_0$ | Start seed of the second logistic map |
| $R$ | Number of rounds |
| $\oplus$ | XOR operation |
| $\|\|$ | Concatenation operation |

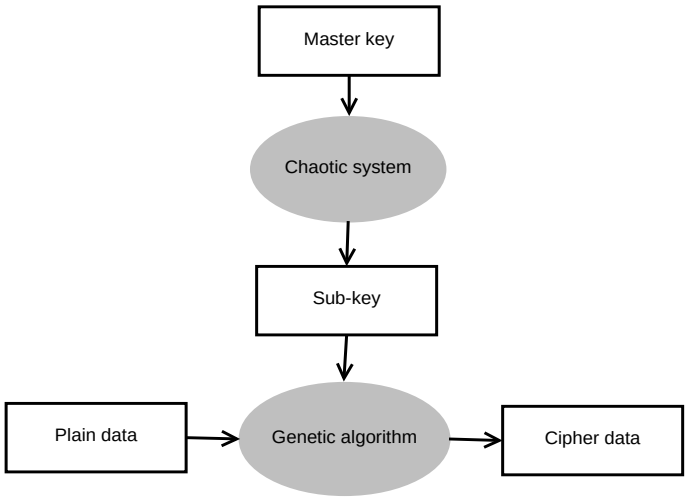


Figure 5.4: Phases of BOSS scheme.

## 5.3 Improved scheme

In this section, we present our improved bio-inspired symmetric encryption scheme named BOSS. The proposed scheme consists of four phases: setup, key schedule, encryption and decryption. It uses 128-bits plain data and 256-bits key to provide 256-bits cipher data. It is based on genetic algorithm and chaotic system as illustrated in Figure 5.4. We adopt crossover and mutation operators for data encryption/decryption. We use a chaotic logistic map to generate the crossover and mutation points. The notations used in our proposed scheme are listed in Table 5.1.

### 5.3.1 Setup

During the setup phase, each constrained IoT device shares a master secret key with the gateway using our previous scheme presented in [121]. Moreover, it is pre-loaded with two logistic maps (See Equation 5.5 and Equation 5.6). We use *round* function as demonstrated by Equation 5.7 and Equation 5.8 to approximate the output of the logistic maps without degradation [180].

$$x_{n+1} = \lambda x_n(1 - x_n) \tag{5.5}$$

$$y_{n+1} = \lambda y_n(1 - y_n) \tag{5.6}$$

$$rx_{n+1} = round(x_{n+1} * 255) \tag{5.7}$$

$$ry_{n+1} = round(y_{n+1} * 127) \tag{5.8}$$

### 5.3.2 Key schedule

During the key schedule phase, the master secret key is expanded based on the first logistic map to generate a sub-key used for one-time encryption. The steps of this phase are shown in Algorithm 1.

**Algorithm 1:** Key schedule of BOSS

**Input** : $SK_i$
**Output** : $k_i$
**BEGIN**
Generate $x_{n+1}$ using Equation 5.5
Generate $rx_{n+1}$ using Equation 5.7
$k_i = SK_i \oplus rx_{n+1}$
**END**

### 5.3.3 Encryption

The encryption phase aims to secure data using genetic operations (*i.e.,* crossover and mutation). In crossover, a part of the plain data and the sub-key are switched. In

mutation, a chosen bit is flipped in both plain data and sub-key. Algorithm 2 details the steps of the encryption phase. The crossover and mutation points are generated using the second logistic map.

After the encryption process, the cipher data are hashed using $HMAC$ to provide data integrity and authenticity which are very necessary properties in IoT environment. The start seeds $x_0$ and $y_0$ are sent only in the first transmission.

**Algorithm 2:** Encryption of BOSS

**Input** : $plaindata$, $SK_i$
**Output** : $cipherdata$
**BEGIN**
Generate a sub-key using Key schedule algorithm
Divide the sub-key $k_i$ into two blocks $\beta_1$ and $\beta_2$ of 128 bits
$\delta = plaindata \oplus \beta_1$
**for** $j = 1, j \leq R, j++$ **do**
  Generate a crossover point $CP$ using Equation 5.6 and Equation 5.8
  $Crossover(\delta, \beta_2, CP)$
  Generate a mutation point $MP$ using Equation 5.6 and Equation 5.8
  $Mutation(\delta, \beta_2, MP)$
**end for**
$cipherdata = (\delta || \beta_2)$
Send$(cipherdata, HMAC(cipherdata||k_i))$
**END**

### 5.3.4 Decryption

When the gateway receives the ciphered data, it generates the sub-key $k_i$ and calculates the $HMAC$. If it is equal to the received $HMAC(cipherdata||k_i)$, it executes the decryption phase using Algorithm 3 to recover the plain data. Otherwise, it rejects the packets.

Since, we propose a symmetric encryption scheme, the decryption process consists of inversing the encryption steps.

## 5.4 Security evaluation

### 5.4.1 Informal security analysis

We present informal security analysis of the proposed scheme based on the assumptions mentioned in the threat model.

**Algorithm 3:** Decryption of BOSS

**Input** : *cipherdata*
**Output** : *plaindata*
**BEGIN**
Divide the *cipherdata* into two blocks $\delta$ and $\beta_2$ of 128 bits
**for** $j = R, j \geq 1, j--$ **do**
  Generate a mutation point $MP$ using Equation 5.6 and Equation 5.8
  $Mutation(\delta, \beta_2, MP)$
  Generate a crossover point $CP$ using Equation 5.6 and Equation 5.8
  $Crossover(\delta, \beta_2, CP)$
**end for**
Divide the sub-key $k_i$ into two blocks $\beta_1$ and $\beta_2$ of 128 bits
$plaindata = \delta \oplus \beta_1$
**END**

**Privacy-preserving**

Privacy-preserving is the ability to protect private information from any disclosure by an attacker. According to Algorithm 2, the sensed data are encrypted using symmetric encryption with 256-bits key. Hence, the probability of an adversary to lunch brute force attack is $1/2^{256}$. Furthermore, the sub-keys that are generated by the logistic chaotic map are used for one-time encryption. Because an adversary cannot reveal the plaindata without the encryption key, privacy-preserving is satisfied. Therefore, the proposed scheme provides privacy-preserving.

**Data integrity**

Integrity is the ability to protect transmitted information from any modification by an attacker. Because we adopt $HMAC$ concept, the gateway can make sure that the message is not modified during the transmission by computing and verifying the hash of the received cipher data. Thus, the proposed scheme ensures data integrity.

**Data authenticity**

Data authenticity is the ability to ensure that the received data was transmitted by a legal sender. During the encrytpion phase, the sub-key is generated from the master key using a chaotic logistic map. The master key is assumed to be shared between the constrained IoT device, controller device and gateway after authentication and key agreement process. The sub-key is combined to the ciphered data and hashed using $HMAC$ function. Hence, the received encrypted data is considered authentic after verification of the $HMAC$. The proposed scheme provides data authenticity.

**Man-in-the-middle attack**

In man-in-the-middle (MITM) attack, the adversary maliciously eavesdrops and probably modifies the transmitted messages between two parties that are directly communicating with each other. Because the privacy-preserving and integrity properties are guaranteed, the proposed scheme resists MITM attack.

**Replay attack**

A replay attack occurs when an adversary eavesdrops on communications and maliciously retransmits the exchanged messages. The transmitted message is encrypted using one-time encryption key. If an attacker replays the same message, the gateway cannot decrypt it due to the key mismatch and thus, it will be discarded. Thus, the proposed scheme overcomes replay attack.

**Denial of service attack**

In DoS attack, the adversary attempts to send fake messages in order to disrupt or interrupt the services of a targeted device. Upon receiving the transmitted message, the received $HMAC$ is verified and only the data from authentic constrained IoT device will be processed. Hence, the proposed scheme is secure against DoS attack.

**Confusion and diffusion**

Confusion conceals the link between the ciphertext and the key. It is difficult to find the key from the ciphertext and if a single bit in a key is changed, most or all bits in the ciphertext will change.

Diffusion conceals the link between the ciphertext and the plaintext. Several or half of the bits in the ciphertext change if a single bit of the plaintext is changed. Likewise, changing one bit in the ciphertext leads to change approximately half of the plaintext bits.

Confusion and diffusion are two cryptographic properties that should be satisfied to provide secure ciphered data and avoid statistical attacks [181]. According to Algorithm 1, the sub-key is generated using chaotic system which is characterized by its sensitivity to the initial conditions and its mixing property. According to Algorithm 2, the cipher data is obtained by applying crossover and mutation functions multiple times on the sub-key and the plain data. The crossover and mutation points are generated using chaotic system. As a result, changing a single bit of the plain data affect the bits of the cipher data. Moreover, an attacker cannot deduce the sub-key from the cipher data. Therefore, the proposed scheme has both high confusion and diffusion.

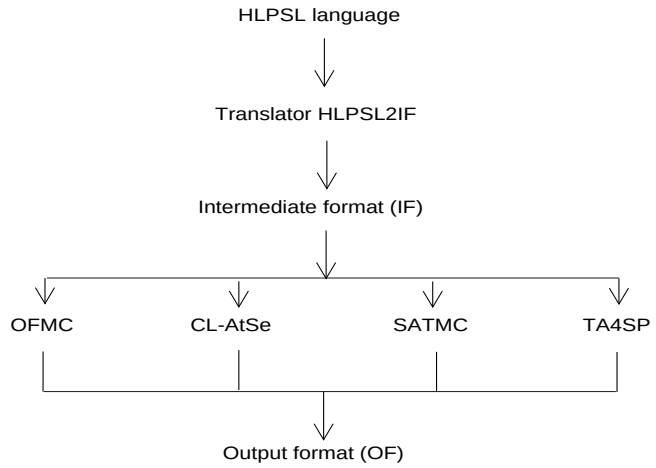


Figure 5.5: AVISPA structure.

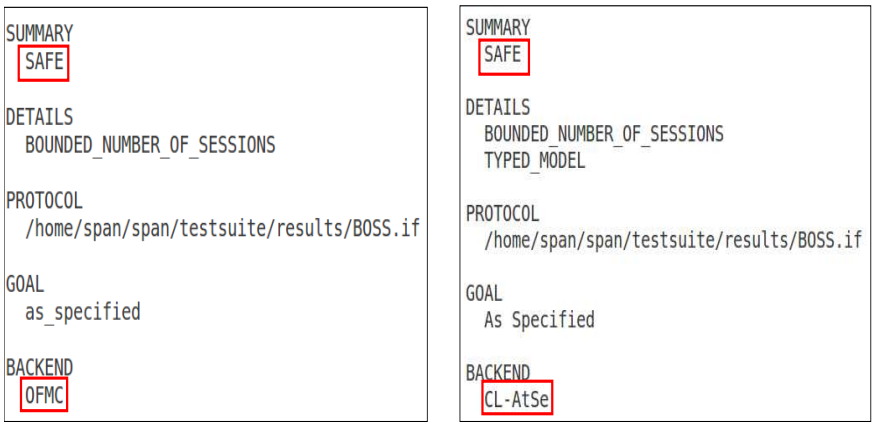


Figure 5.6: Formal security verification results of BOSS scheme.

### 5.4.2 Formal security verification using AVISPA

AVISPA is a widely accepted software that checks if a security protocol is $SAFE$ or $UNSAFE$ against active and passive attacks. The security protocol is specified using high level protocol specification language (HLPSL) and can be verified under four back-ends; on-the-fly model-checker (OFMC), SAT-based model-checker (SATMC), constraint logic-based attack searcher (CL-AtSe) and tree automata based on automatic approximations for the analysis of security protocols (TA4SP) [163–165]. The OFMC and CL-AtSe are the most used models. Figure 5.5 shows the AVISPA tool's structure.

The formal verification result of our proposed scheme using the OFMC and CL-AtSe models is demonstrated in Figure 5.6. Our scheme is $SAFE$ under OFMC and CL-AtSe and thus, it is secure against active and passive attacks.

Table 5.2: Simulation parameters.

| **Parameter** | **Value** |
|---|---|
| Operating system | Ubuntu 16.04 LTS |
| Network area | 100m×100m |
| Gateway position | (100,50) |
| Controller position | (50,50) |
| Number of sensors | 10, 20, 30 |
| Distance between sensors | 10m |
| Initial energy of sensors | 2J |
| Communication range of sensors | 50m |
| Interval between packets | 10s |
| $E_{elec}$ | $50nJ/bit$ |
| $\epsilon_{fs}$ | $10pJ/bit/m^2$ |
| $\epsilon_{mp}$ | $0.0013pJ/bit/m^4$ |
| Simulation time | 1800s |

## 5.5 Simulation results

In order to study the efficiency and the practicality of our proposed scheme, we simulated it using network simulator 3 (NS-3). NS-3 is open-source and widely used tool for discrete event network simulations. It supports several models such as TCP/UDP, IPv4, WiFi, etc and includes many libraries to provide different types of networks. The simulation programm can be written using $C++$ or *Python* scripts [182]. Table 5.2 illustrates the simulation parameters considered in our proposed scheme. We consider three different scenarios; scenario 1 with 10 sensors, scenario 2 with 20 sensors and scenario 3 with 30 sensors. Simulation results in terms of packet delivery ratio, packet delay, throughput and energy consumption are discussed in next sub-sections.

### 5.5.1 Packet delivery ratio

Packet delivery ratio is the ratio of total received packets to the total sent packets. The values of packet delivery ratio of the proposed scheme under the three considered scenarios are shown in Figure 5.7. For scenario 1 and 2, the controller device receives 1800 and 3600 packets, respectively (*i.e.,* all sent packets are received). However, it receives 5220 packets instead of 5400 for scenario 3. Therefore, the packet delivery ratio decreases when the number of IoT devices increases. Because a large number of messages are sent and the IoT devices send the encrypted data at random times, some

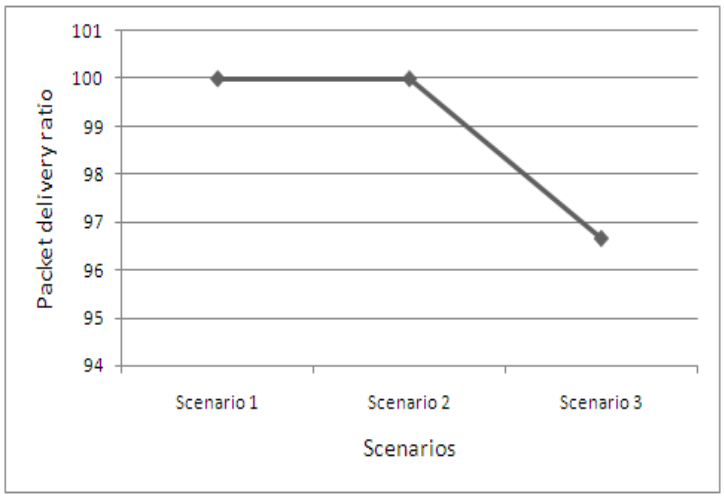


Figure 5.7: Packets delivery of BOSS scheme.

of the transmitted packets may be lost and the controller device can not receive these packets.

### 5.5.2 Packet delay

Packet delay is the transmission time of all sent packet. Figure 5.8 demonstrates the values of packet delay of our proposed scheme under scenario 1, scenario 2 and scenario 3. The values of packet delay are 4.15ms , 4.45ms and 5.11ms for scenario 1, 2 and 3, respectively. We observe that the packet delay increases if the number of IoT devices increases as well. With the increasing number of IoT devices, more messages are exchanged and thus, the reception time of all packets will increase. However, our scheme has a small increased packet delay from scenario 1 to 2 and from scenario 2 to 3.

### 5.5.3 Throughput

Throughput is the number of transmitted bits per unit of time. The values of throughput in bytes per seconds (bps) of our proposed scheme under the three different scenarios are given in Figure 5.9. When the number of IoT devices increases, the total time of exchanging messages increases as well. In scenario 1, 2 and 3 the amount of data transmitted in the network are 12502bps, 11675bps and 10174bps, respectively. As a result, our scheme has high throughput even if the values decrease by 6% from scenario 1 to 2 and by 12% from scenario 2 to 3. This decrement does not affect the

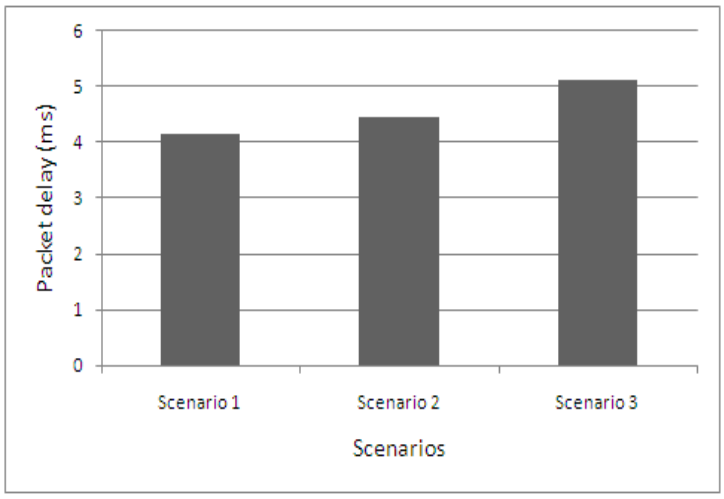


Figure 5.8: Packets delay of BOSS scheme.

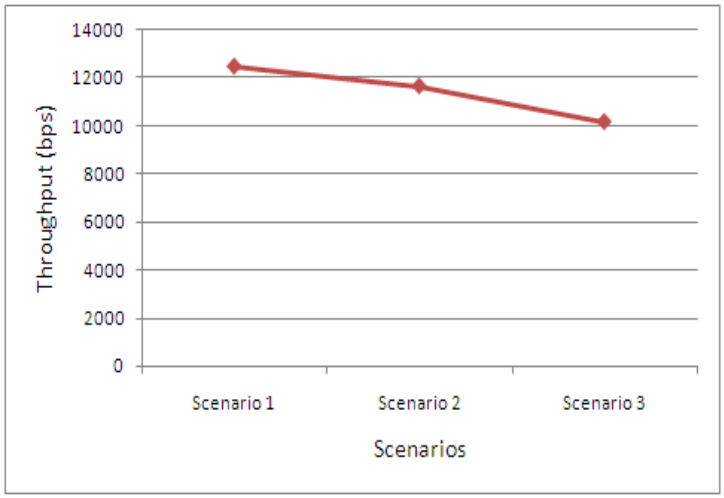


Figure 5.9: Throughput of BOSS scheme.

performance of our proposed scheme.

### 5.5.4 Energy consumption

According to the energy model described previously, the energy consumption depends on the size of the transmitted/received message and the distance to the destination node. Figure 5.10 shows the consumed energy of constrained IoT devices in our proposed scheme during the encryption phase. Each IoT device consumes a small amount of energy (0.0312 mJ) to send one encrypted data to the controller device. This is due to the less communication overhead incurred during the encryption phase. Therefore, our scheme is an efficient solution for data encryption in WSN-enabled IoT where the energy consumption is a crucial issue.

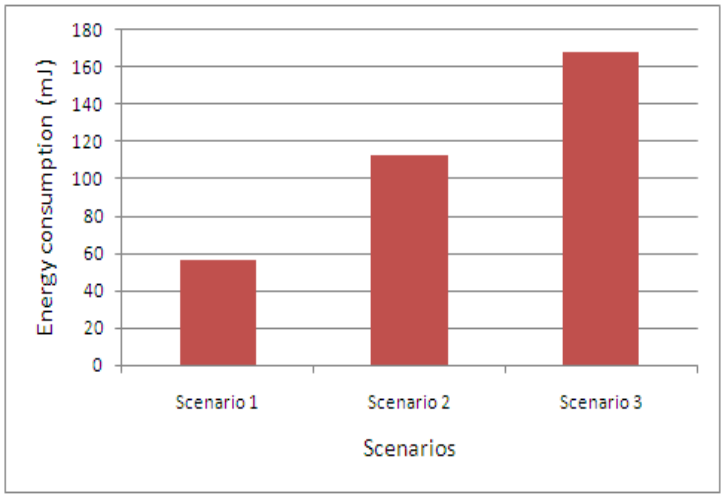


Figure 5.10: Energy consumption in BOSS scheme.

## 5.6 Comparative analysis

In this section, we provide a comparative analysis of our proposed scheme with Aljawarneh et al.'s scheme [139] and related methods [138,140] in terms of computation, communication and storage cost and security requirements.

To calculate the total computational time for each scheme, we used JAVA language on Intel Core i3-4005U@1.70GHz to define the following cryptographic operations' cost:

— Time consumed by the hash function $T_H = 608ms$.

— Encryption time using AES algorithm $T_{AES} = 655ms$.

— Encryption time using RSA algorithm $T_{RSA} = 312ms$.

— Encryption time using genetic algorithm $T_G = 16ms$.

We ignore the computational time of feistel algorithm since it is based on inexpensive operations (*i.e.,* shift and logical operations). Table 6.3 summarizes the computational overheads related to the encryption phase. Compared to Aljawarneh et al.'s scheme [139] and recent related works [138, 140], our proposed scheme has clearly low computational cost because the encryption process requires lightweight operations such as crossover and mutation to encrypt the collected data. However, the scheme presented in [138–140] are based on conventional algorithm such as AES and RSA that involve large time to perform the data encryption.

For comparison of communication cost, we assume that the length of the plaindata is 128 bits, the output of the hash function is 160 bits and the crossover/mutation point is 8 bits. From Table 6.4, our proposed scheme has a little higher communication cost

Table 5.3: Comparison of computational cost of BOSS scheme.

| **Schemes** | **Computational cost** | **Computational time(ms)** |
|---|---|---|
| Aljawarneh et al. [139] | $T_{AES} + T_G$ | 671 |
| Song et al. [138] | $T_{AES} + T_H$ | 1263 |
| Elhoseny et al. [140] | $T_{AES} + T_{RSA} + 2T_H$ | 2183 |
| BOSS | $T_G + T_H$ | 624 |

Table 5.4: Comparison of communication cost of BOSS scheme.

| **Schemes** | **Communication cost (bits)** |
|---|---|
| Aljawarneh et al. [139] | $128 + 160$ |
| Song et al. [138] | $128 + 160$ |
| Elhoseny et al. [140] | $160 + 160 + 128$ |
| BOSS | $256 + 160$ |

compared to Aljawarneh et al.'s scheme [139] and Song et al.'s scheme [138] because the cipherdata in [139] and [138] is of the same length as the plaindata. However, in our scheme the encryption algorithm outputs a cipherdata of $2n$ bits. In Aljawarneh et al.'s scheme [139], the 160 bits represent the crossover and mutation points. Although, in our scheme the added 160 bits are the output of hash function that provides the data integrity. On the other hand, our proposed scheme BOSS has less communation cost compared to Elhoseny et al.'s scheme [140].

The storage cost of our proposed scheme is produced by storing the encryption key, the cipherdata and the hashed data which is $416 + 2n$ bits. In contrast, the schemes recently presented in [138–140] require large storage space because they are based on conventional encryption algorithms that cannot be implemented on resource-constrained devices as demonstrated in [183].

Table 5.5 provides the security features of our proposed scheme and the recent related schemes presented in [138–140]. We observe that our scheme BOSS achieves

Table 5.5: Comparison of security features of BOSS scheme.

| **Security features** | **[139]** | **[138]** | **[140]** | **BOSS** |
|---|---|---|---|---|
| Data confidentiality | + | + | + | + |
| Data integrity | - | + | - | + |
| Data authenticity | - | + | - | + |
| Replay attack | - | + | - | + |
| DoS attack | - | + | - | + |
| MITM attack | - | + | - | + |
| Avalanche effect | 52.5% | / | / | 63% |
| Key space | $2^{128}$ | $2^{256}$ | $2^{128}$ | $2^{256}$ |

+ achieved, - not achieved, / not mentionned

fundamental security properties such as data confidentiality, integrity and authenticity. Additionally, it resists various attacks and has greater avalanche effect and key space than Aljawarneh et al.'s scheme [139] and Elhoseny et al.'s scheme [140]. Therefore, our proposed scheme BOSS satisfies the security requirements for smart IoT environments.

## 5.7 Conclusion

In this chapter, we proposed a lightweight bio-inspired security scheme based on GA and chaotic system. We adopted a symmetric encryption technique to encrypt the collected data before the data transmission process. Formal and informal security analyses showed that the proposed scheme resists active and passive attacks, and satisfies data privacy, integrity and authenticity with high confusion and diffusion. Simulation results using NS-3 showed that the proposed scheme is suitable for IoT with constrained-ressource devices. Compared to Aljawarneh et al.'s scheme and recent encryption approaches, our proposed scheme is more efficient in terms of computation and storage costs. Furthermore, it is more secure and resilient to known IoT security attacks.

In next chapter, we extend our work to enable end-user devices to securely access the encrypted data in the cloud server using blockchain technology.

# Chapter 6

# Lightweight blockchain-based remote user authentication for fog-enabled IoT deployment

## 6.1 Introduction

The Internet of things (IoT) has been recognized as new era of Internet; it consists of everyday physical objects interconnected to collect and share data through the Internet. Currently, 50 billions of devices are expected to be connected to the Internet [3]. These devices exchange a large amount of data and autonomously provide smart services to improve human life. The majority of IoT devices are inherently resource-constrained; they have limited processing and storage capabilities.

Cloud computing is a promising technology that solves big data problems in IoT by proliferating hardware ressources' abilities. It is an effective solution to process and store data generated by IoT devices. This allows remote users to access the collected data stored on a cloud server and make relevant decisions [184,185]. Fog computing is a distributed infrastructure that extends cloud computing services to the end-users. For instance, a healthcare professional can access patient's data stored on the hospital cloud server through fog nodes to reduce delay and latency. However, it is very necessary to authenticate remote users before accessing the cloud data [43].

Remote user authentication is a crucial task to verify the legitimacy of end-users over open networks and establish a session key that will be used to transmit cloud data

securely. Different factors such as identity/username, password and biometrics can be used to authenticate remote users [156, 186].

The proposed schemes presented in [141–143, 145, 146, 148] are based on centralized architecture and thus, they are limited in terms of scalability, availability and security since they are prone to single point of failure. On the other hand, the schemes proposed in [144, 147] use blockchain technology to achieve user authentication. However, they did not consider the constrained resources of end-user devices and are inefficient in terms of computation overhead. Moreover, they have several security weaknesses such as lack of privacy-preserving and session key agreement.

The present chapter presents a lightweight blockchain-based scheme to provide user authentication in fog-enabled IoT systems. The proposed scheme is based on blockchain technology and fog computing and uses a lightweight cryptographic hash function to address the limitations of previous works. We formally verified the security of our proposed scheme using AVISPA tool and implement it using solidity language. A comparative analysis is also provided to show its efficiency and robustness.

## 6.2 Preliminaries

### 6.2.1 Smart contract

Smart contract is a computer program stored on the blockchain and automatically executed when triggered by a transaction. It is based on predefined rules/functions and generates events that will be broadcasted to all participating nodes. The execution cost of smart contract within Ethereum blockchain is called *gas* [187, 188]. It is used to reward miners for the computational resources to execute smart contracts.

In our work, we use public blockchain with smart contract instead of private blockchain to support scalability and make the system open to any user. The key benefits of using blockchain with smart contract to provide remote user authentication in IoT systems are listed below.

— **Decentralization** : the decentralized architecture makes the authentication solution more scalable and solves the single point of failure problem. Moreover, it can resist relevant security attacks such as denial of service attack.

— **Immutability** : the collected data of IoT devices can only be accessed by authenticated users. The blockchain allows credentials of legitimate users to be stored securely and prevents the modification of data and user impersonation by attackers.

— **Anonymity and privacy** : the user information stored on the blockchain do not reveal the real identity of users and thus, anonymity and privacy are preserved.

— **Secure authentication** : the authentication process is executed by smart contract (*i.e.,* without any possibility of fraud or adversary disruption) which disables malicious users from authorized access to data.

### 6.2.2 Hash function

Hash function is one-way cryptographic function defined by $h : \{0,1\}^* \rightarrow \{0,1\}^n$, it generates a fixed length output (of $n$ bits) from an arbitrary length binary string. The result or output of hash function is called message digest or hash value; it is used to ensure data integrity [159]. The main properties of one-way hash functions are defined as follows.

— **Lightweight computation** : the hash function is a lightweight cryptographic technique that takes fast and easy steps to generate the message digest.

— **Difficulty of inversion** : given a message digest $y$, it is difficult to find $x$ such that $h(x) = y$.

— **Collision resistant** : it must be difficult to find two binary strings $x_1$ and $x_2$ whose hash values $h(x_1)$ and $h(x_2)$ are the same.

### 6.2.3 Fuzzy extractor

Fuzzy extractor is a widely-accepted technique that uses biometric data to generate strong cryptographic keys. It is based on two algorithms [189].

**Gen** : is a probabilistic algorithm defined by $Gen(B) = (\sigma, \tau)$, it takes biometric data $B$ as input and generates biometric secret key $\sigma$ and public reproduction parameter $\tau$.

**Rep** : is a deterministic algorithm defined by $Rep(B^*, \tau) = \sigma$, it reproduces the original secret key $\sigma$ if the Hamming distance between $B^*$ and $B$ is less than or equal to a predefined error tolerance threshold value $t$.

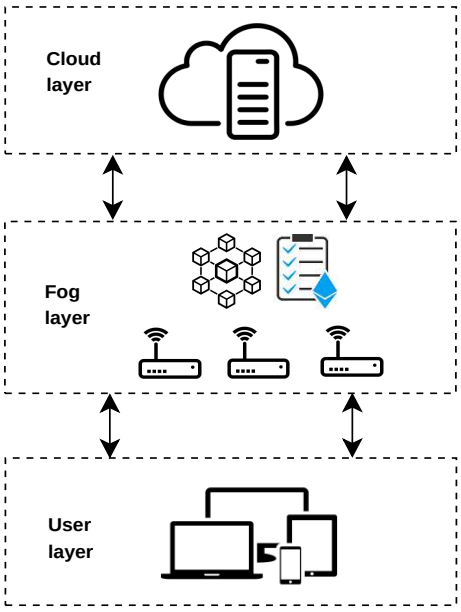


Figure 6.1: Network architecture of Lightchain scheme.

### 6.2.4 Network model

In order to allow remote users to access to cloud data in a trustworthy way, we consider a three-layered architecture that includes user layer, fog layer and cloud layer. Figure 6.1 illustrates our network architecture.

— User layer : This layer mainly consists of remote users' devices that require to access to data generated by IoT devices. These data are securely stored on a cloud server.

— Fog layer : This layer contains several fog nodes deployed on the edge of the network and associated with a public blockchain. These fog nodes communicate with each other to execute the consensus algorithm. A smart contract is defined on the blockchain to provide distributed user authentication. The fog layer represents the security layer between end-users and cloud server.

— Cloud layer : This layer includes a cloud server that processes and stores the data collected by IoT devices. It is the main component in our network. The security of cloud server is out of topic in this paper, it is assumed to be secure and trustworthy.

### 6.2.5 Threat model

We consider the widely-used Dolev Yao threat model [179] where an adversary $\mathcal{A}$ can intercept, modify or re-send the transmitted messages over insecure channels. However, $\mathcal{A}$ cannot reveal any information transmitted through secure channels. In addition, we assume that $\mathcal{A}$ can impersonate a legitimate user or fog node in order to be authenticated by the system. However, $\mathcal{A}$ cannot impersonate or compromise the cloud server because it is assumed to be well secured and trusted. We also suppose that if $\mathcal{A}$ knows $B = C \oplus D$, he/she cannot find $C$ and $D$ in polynomial time. Finally, the user smart device can be stolen by $\mathcal{A}$ and the information stored on its memory can be extracted using power analysis [190].

## 6.3 Proposed scheme

The proposed scheme called Lightchain aims to provide secure and lightweight remote user authentication using blockchain and hash function. Each remote user is authenticated by a fog node to access the cloud server's data securely. It consists of three phases including initialization, user registration and user login and authentication phase. All notations used in our proposed scheme are listed in Table 6.1.

### 6.3.1 Initialization

This phase is executed by the system administrator (SA) in an offline mode. Initially, the SA selects two secret keys $X$ and $Y$ and chooses a unique identity $ID_j$ for each fog node. Then, he/she computes the credential of fog node $CF_j = h(h(ID_j)||X)$. Finally, $(ID_j, CF_j, Y)$ and $(h(ID_j), CF_j)$ are stored on fog node and cloud server, respectively.

### 6.3.2 User registration

During this phase, a user $U_i$ can only register to one fog node using the following steps, as illustrated in Figure 6.2.

- $U_i$ selects a random number $r_i$, an identity $ID_i$ and a password $PW_i$.
- $U_i$ calculates $MID_i = h(ID_i||r_i)$ and $MPW_i = h(PW_i||r_i)$.

Table 6.1: Notations used for Lightchain scheme.

| **Notation** | **Description** |
|---|---|
| $SA$ | System administrator |
| $CS$ | Cloud server |
| $FN_j$ | $j^{th}$ fog node |
| $U_i$ | $i^{th}$ user |
| $X$, $Y$ | Two large secret keys |
| $ID_j$ | Identity of the $j^{th}$ fog node |
| $ID_i$ | Identity of the $i^{th}$ user |
| $CF_j$ | Credential of $FN_j$ |
| $TT_i$ | Trust token of $U_i$ |
| $r_i$, $r_j$, $r_{CS}$ | Random number generated by $U_i$, $FN_j$ and $CS$, respectively |
| $h()$ | One-way hash function |
| $Gen()$, $Rep()$ | Fuzzy extractor functions |
| $\sigma_i$ | Biometric secret key of $U_i$ |
| $\tau_i$ | Public parameter of $U_i$ |
| $T_1$, $T_2$, $T_3$, $T_4$ | Current timestamps |
| $\Delta T$ | Maximum transmission delay |
| $SK_i$ | Secret session key shared between $U_i$ and $CS$ |
| $\oplus$, $\|\|$ | XOR and concatenation operations |

- $U_i$ sends $(MID_i, MPW_i)$ to the nearest fog node $FN_j$ via a secure channel.
- $FN_j$ checks the existence of $MID_i$ and $MPW_i$ in the blockchain using the smart contract. Then, it computes $A_i = h(MID_i||Y)$, $B_i = h(MPW_i||Y)$ and user trust token $TT_i = h(A_i||B_i)$. After that, a block of mapping between $(MID_i, MPW_i, TT_i)$ is created. Algorithm 4 describes this step.
- $FN_j$ sends $TT_i$ to $U_i$ via a secure channel.
- $U_i$ inputs his/her personnal biometric $BIO_i$ and computes $Gen(BIO_i) = (\sigma_i, \tau_i)$, $D_i = h(TT_i||\sigma_i)$ and $E_i = r_i \oplus h(ID_i||PW_i)$.
- $U_i$ stores $(TT_i, D_i, E_i, \tau_i)$.

### 6.3.3 User login and authentication

Once the registration phase is successfully finished, the user is required to be authenticated by the fog node using the following steps. Figure 6.3 summarizes the user login and authentication phase.

- $U_i$ inputs his/her identity $ID_i$, password $PW_i$ and personal biometric $BIO_i$ on his/her smart device.

**Algorithm 4:** User registration for smart contract

**Parameters** :
*lightchain* : Blockchain
*user* : Object
**Begin**
**if** (IdExists(user.mid, lightchain) = $false$) **then**
**if** (PwExists(user.mpw, lightchain) = $false$) **then**
CreateMapping(user.mid, user.mpw, user.tt, lightchain)
**end if**
**else**
return Error()
**end if**
**End**

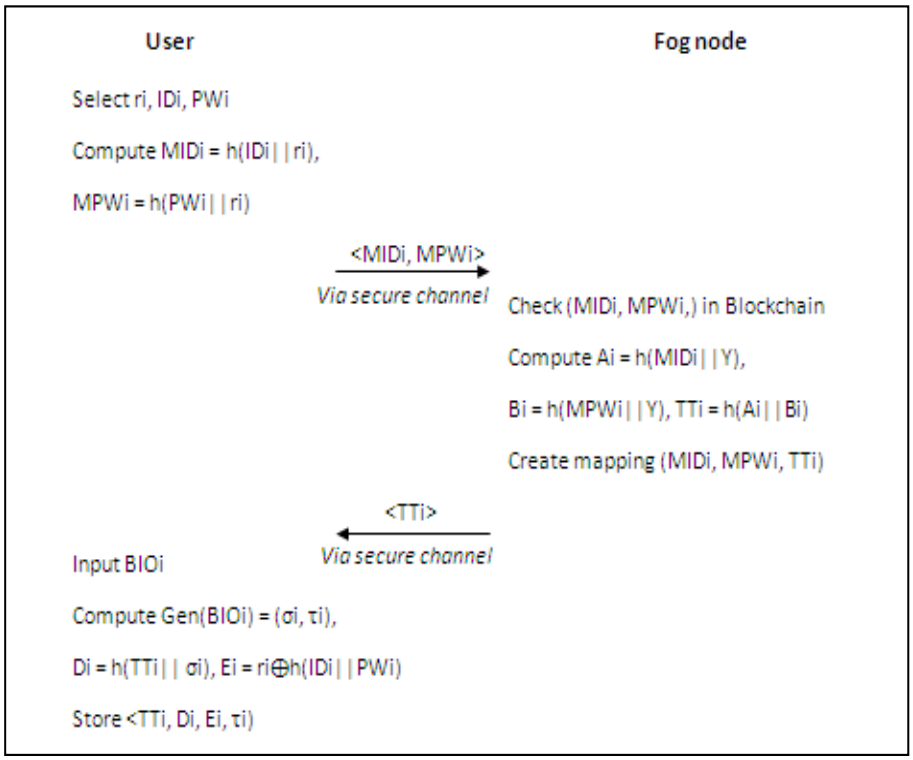


Figure 6.2: Registration phase of Lightchain.

- The user smart device computes $\sigma_i = Rep(BIO_i, \tau_i)$ and $D_i^* = h(TT_i||\sigma_i)$, then checks if $D_i^* = D_i$. If the verification holds, it executes the next steps. Otherwise, it rejects the login request.
- $U_i$ computes $r_i = E_i \oplus h(ID_i||PW_i)$, $MID_i = h(ID_i||r_i)$, $MPW_i = h(PW_i||r_i)$, $M_1 = (MID_i||MPW_i)$, $M_2 = h(M_1) \oplus TT_i$ and $M_3 = h(M_2||T_1)$ where $T_1$ is the user current timestamp.
- $U_i$ sends $(M_1, M_2, M_3, T_1)$ to the nearest fog node via a public channel.
- $FN_j$ receives the authentication request at time $T_1^*$ and checks if $|T_1^* - T_1| \leq \Delta T$. If the verification holds, it checks the integrity of the received message. Otherwise, it rejects the authentication request.
- $FN_j$ computes $M_3^* = h(M_2||T_1)$ and checks if $M_3^* = M_3$. If the verification holds, it executes the next steps. Otherwise, it rejects the authentication request.
- $FN_j$ computes $TT_i^* = M_2 \oplus h(M_1)$ then extracts $MID_i$, $MPW_i$ from $M_1$ and checks the existence of $MID_i$ and $MPW_i$ in the blockchain. Moreover, it checks the validity of mapping $(MID_i, MPW_i, TT_i)$ and the trust token $TT_i^*$ is compared to the one stored for the same device during the registration phase. Algorithm 5 describes this step.
- $FN_j$ generates a random number $r_j$, a timestamp $T_2$ and computes $M_4 = h(MID_i||MPW_i||r_j)$ and $M_5 = h(h(ID_j)||CF_j||T_2) \oplus r_j$.
- $FN_j$ sends $(M_1, h(ID_j), M_4, M_5, T_2)$ to the CS via a public channel.
- The CS receives the access request at time $T_2^*$ and checks if $|T_2^* - T_2| \leq \Delta T$. If the verification holds, it executes the next steps. Otherwise, it rejects the access request.
- The CS computes $r_j^* = M_5 \oplus h(h(ID_j)||CF_j||T_2)$ and $M_4^* = h(M_1||r_j^*)$. If $M_4^* = M_4$, the fog node is authenticated and the process continues. Otherwise, the fog node is considered as malicious.
- The CS chooses a random number $r_{CS}$, generates a timestamp $T_3$ and calculates $SK_i = h(M_1||r_{CS})$, $M_6 = h(M_1||T_3) \oplus r_{CS}$ and $M_7 = h(SK_i||M_6)$.
- The CS sends $(M_6, M_7, T_3)$ to the fog node via a public channel.
- $FN_j$ receives the access response and verifies if $|T_3^* - T_3| \leq \Delta T$. Then, it computes

$M_8 = M_7 \oplus TT_i$, generates a timestamp $T_4$ and sends $(M_6, M_8, T_3, T_4)$ to the user via a public channel.

- $U_i$ receives the authentication response at time $T_4^*$ and checks if $|T_4^* - T_4| \leq \Delta T$. If the verification holds, it executes the next steps. Otherwise, it rejects the authentication response.
- $U_i$ computes $r_{CS}^* = M_6 \oplus h(MID_i||MPW_i||T_3)$, $SK_i^* = h(MID_i||MPW_i||r_{CS})$, $M_7^* = h(SK_i^*||M_6)$ and $M_8^* = M_7^* \oplus TT_i$. If $M_8^* = M_8$, the fog node is authenticated and the user can securely communicate with the CS using $SK_i^*$. Otherwise, he/she cannot access the data stored on the CS.

**Algorithm 5:** User authentication for smart contract

```
Parameters :
        lightchain : Blockchain
        user : Object
Begin
if (IdExists(user.mid, lightchain) = true) then
  if (PwExists(user.mpw, lightchain) = true) then
    if (ValidateMapping(user.mid, user.mpw, user.tt, lightchain) = true) then
      User is successfully authenticated
    end if
  end if
else
  return Error()
end if
End
```

## 6.4 Security evaluation

### 6.4.1 Informal security analysis

**User impersonation attack**

In user impersonation attack, an adversary $\mathcal{A}$ impersonates the identity of a legitimate user to be authenticated by a fog node. Suppose that $\mathcal{A}$ tries to send an authentication request, he/she needs to know $MID_i$, $MPW_i$ and $TT_i$. Even if $\mathcal{A}$ stoles the smart device of a user $U_i$, he/she cannot send an authentication request since he/she needs personal biometrics of $U_i$ to validate the login and continue the process of authentication. Therefore, our Lightchain is secure against user impersonation attack.

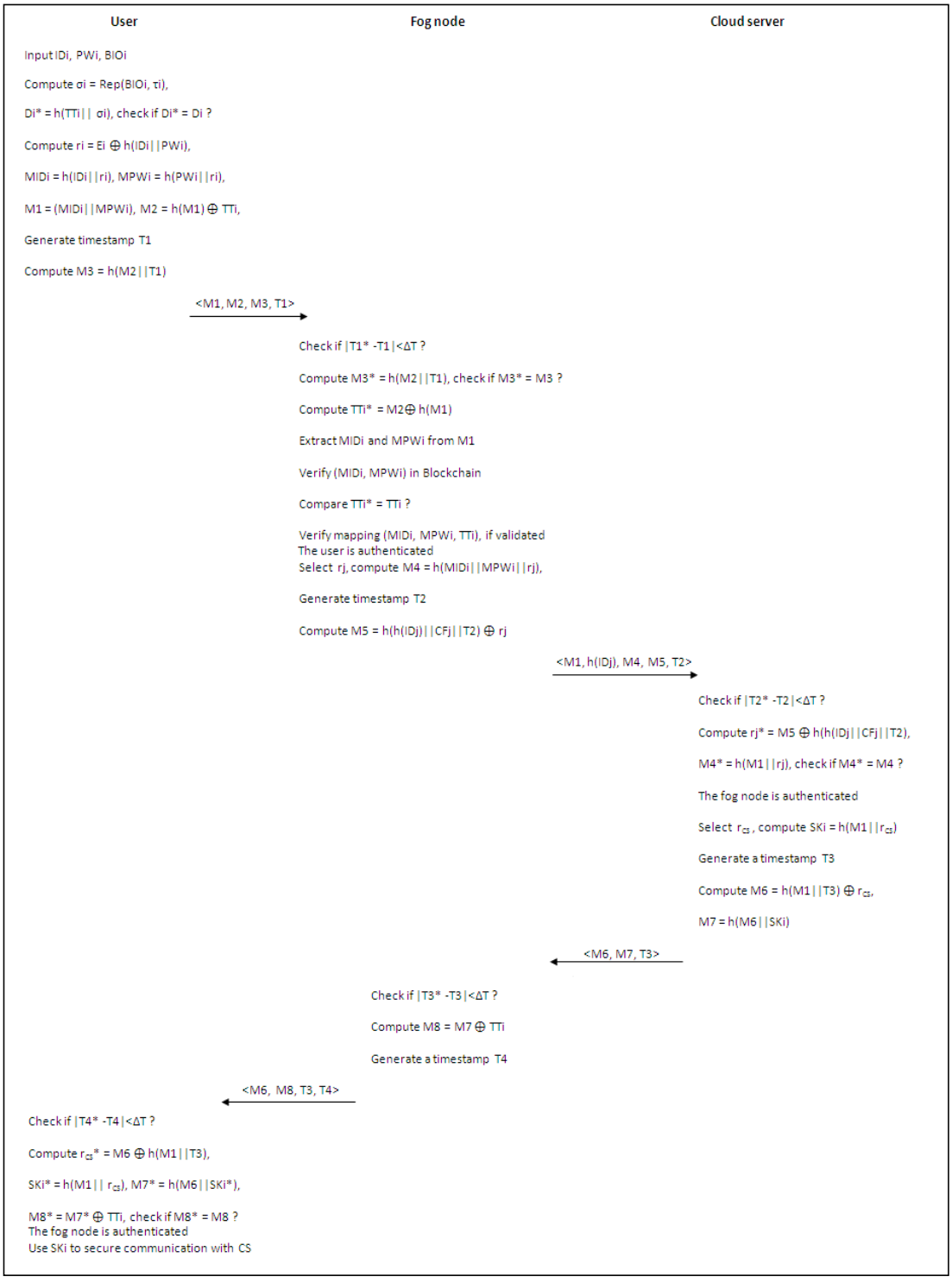


Figure 6.3: Login and authentication phase of Lightchain.

**Fog node impersonation attack**

An attacker $\mathcal{A}$ may impersonate a fog node by sending valid access request and response request to the $CS$ and $U_i$, respectively. However, $\mathcal{A}$ needs to guess the identity $ID_j$ and credential $CF_j$ of fog node and the shared trust token $TT_i$. Hence, our proposed scheme is protected from fog node impersonation attack.

**Privileged insider attack**

In this attack, we assume that an insider user $\mathcal{A}$ has the registration information $MID_i$ and $MPW_i$ sent by the user $U_i$ to the fog node. Moreover, we suppose that $\mathcal{A}$ has the smart device of $U_i$ and can read the stored information using power analysis. However, the attacker is unable to provide personal biometric of $U_i$ required to validate the login. As a result, our proposed Lightchain resists the insider attack.

**Identity and password guessing attack**

We assume that the smart device of $U_i$ is stolen by an adversary $\mathcal{A}$ which can retrieve the stored information $(TT_i, D_i, E_i, \tau_i)$. However, $\mathcal{A}$ cannot guess the identity $ID_i$ and password $PW_i$ without the knowledge of $r_i$. Furthermore, the secret random number $r_i$ is not directly stored on the user device and is not sent via public channels. Thus, our Lightchain is resilient to identity and password guessing attack.

**Stolen user device attack**

If the user smart device is stolen by an adversary $\mathcal{A}$, the authentication of $\mathcal{A}$ remains impossible because he/she needs the secret biometric key $\sigma_i$ and the secret random number $r_i$. Hence, our proposed scheme is secure against stolen user device attack.

**Replay attack**

In replay attack, an attacker $\mathcal{A}$ maliciously intercepts transmitted messages and tries to retransmit them in order to be authenticated as a legitimate entity. However, in our scheme all sent messages include a timestamp which is verified on the receiver. Moreover, even if $\mathcal{A}$ alters the timestamp, the receiver checks the integrity of the message and only accepts the non-modified ones. Therefore, our proposed scheme is protected against replay attack.

**Man in the middle (MIM) attack**

In MIM attack, the adversary $\mathcal{A}$ secretly listens to the communications and possibly modifies or replays the transmitted messages. However, in our scheme all messages are protected with one-way hash function and thus $\mathcal{A}$ cannot reveal or change the original

data. Moreover, our proposed scheme is secure against replay attack. As a result, it resists MIM attack.

**Denial of service (DoS) attack**

In our proposed scheme, we assumed that the CS is well secured by a strong IDS system. Furthermore, we deploy smart contract on fog nodes to authenticate legal users. If a malicious adversary $\mathcal{A}$ sends several authentication requests, the smart contract will discard such requests and only the data of authentic users will be processed. Thus, our Lightchain is resilient to DoS attack.

**Mutual authentication**

During user login and authentication phase, the user $U_i$ and the fog node $FN_j$ are mutually authenticated. $FN_j$ authenticates $U_i$ using the smart contract that verifies the association $(MID_i, MPW_i, TT_i)$. Correspondingly, $U_i$ authenticates $FN_j$ if only the verification of $M_8$ is successful. Hence, our proposed scheme achieves mutual authentication.

**Session key secrecy**

In our Lightchain, a session key $SK_i$ is calculated between the authenticated user and the CS. The computation of $SK_i$ depends on the random number $r_{CS}$. In addition, a session key is only used for one communication between $U_i$ and $CS$. As a result, the proposed Lightchain satisfies the session key secrecy.

**User anonymity**

In the proposed scheme, the user $U_i$ hides its identity $ID_i$ using a random number $r_i$ and one-way hash function before data transmission. Even if an adversary $\mathcal{A}$ intercepts communications on public channels, it is computationally infeasible to reveal $ID_i$. Therefore, our scheme preserves user anonymity.

**Data privacy**

During the registration phase, the remote user sends masked identity and password to the fog node that executes hash function to store the user data securely. Furthermore, after the login and authentication phase, the authenticated user can securely access the data stored on the CS. The shared session key $SK_i$ is used to secure the communications between $U_i$ and $CS$. Hence, data privacy is well protected in our proposed scheme.

**Data integrity**

An adversary $\mathcal{A}$ can intercept and alter the transmitted messages. However, in our

scheme all messages are hashed using one-way hash function. Each request contains the data and its hash value. The receiver verifies the hash value and determines whether the message is modified or not. Thus, our proposed scheme guarantees data integrity.

**Data availability**

In cloud-based IoT applications, data availability is a main issue that needs to be addressed. In our proposed scheme, after the login and authentication phase, the data stored on the CS is accessible to legitimate users. Moreover, we assume that the CS is secure and well protected. Therefore, our scheme provides data availability.

### 6.4.2 Formal security verification using AVISPA

In this section, we formally verify our proposed scheme using automated validation of Internet security protocols and applications (AVISPA) tool. AVISPA is a widely-accepted tool that validates the security of protocols against active and passive attacks such as replay and MTM attacks. It is based on a role language called high level protocol specification language (HLPSL) which specifies the intended security properties of the protocol. AVISPA also integrates four back-ends: on-the-fly model-checker (OFMC), SAT-based model-checker (SATMC), constraint logic-based attack searcher (CL-AtSe) and tree automata based on automatic approximations for the analysis of security protocols (TA4SP) that analyze and verify the security of protocols [163–165].

To specify the intended security properties of our proposed scheme related to login and authentication phase, we initially implemented three basic roles for end-user device, fog node and cloud server. Then, we specified the role for session that instantiates the basic roles with concrete arguments and the role for environment that contains global constants and composition of sessions. Finally, we identified the goals of our proposed scheme which represents the intended security properties.

To fomally verify the security of our proposed scheme using AVISPA tool, we used security protocol animator for AVISPA (SPAN) simulator under OFMC model. The OFMC is the most used model to analyze and validate security protocols using AVISPA tool [191]. It supports algebraic operations and verifies the correctness of the protocol. The OFMC back-end executes several symbolic techniques to detect attacks based on Dolev-Yao model.

The verification results presented in Figure 6.4 indicate that the security goals of

```
% OFMC
% Version of 2006/02/13
SUMMARY
  SAFE
DETAILS
  BOUNDED_NUMBER_OF_SESSIONS
PROTOCOL
  /home/span/span/testsuite/results/Lightchain.if
GOAL
  as_specified
BACKEND
  OFMC
```

Figure 6.4: Formal security verification results of Lightchain scheme.

the proposed scheme are satisfied and our scheme is SAFE under OFMC back-end. Therefore, it is secure against active and passive attacks.

## 6.5 Implementation results

We used Ethereum blockchain with smart contract to authenticate remote users. In this section, we develop the smart contract using Solidity language under Remix IDE [1]. Solidity is a high-level programming language for the implementation of smart contract. Remix is a user-friendly and powerful IDE for compilation and deployment of Ethereum smart contracts. It integrates a local blockchain and provides multiple functions to allow smart contracts deployment and interaction. Initially, we test the smart contract operations including user registration and authentication with different scenarios. Subsequently, we consider three key IoT use cases to evaluate the execution cost and financial cost of the proposed smart contract.

### 6.5.1 Test scenarios

When a new user sends a registration request to the nearest fog node, the smart contract generates a successful registration event as shown in Figure 6.5. The registered user can access the data in the cloud by sending an authentication request. In this scenario, the smart contract verifies the proper mapping and generates a successful authentication event, as presented in Figure 6.6. However, if a non-registered user

1. remix.ethereum.org

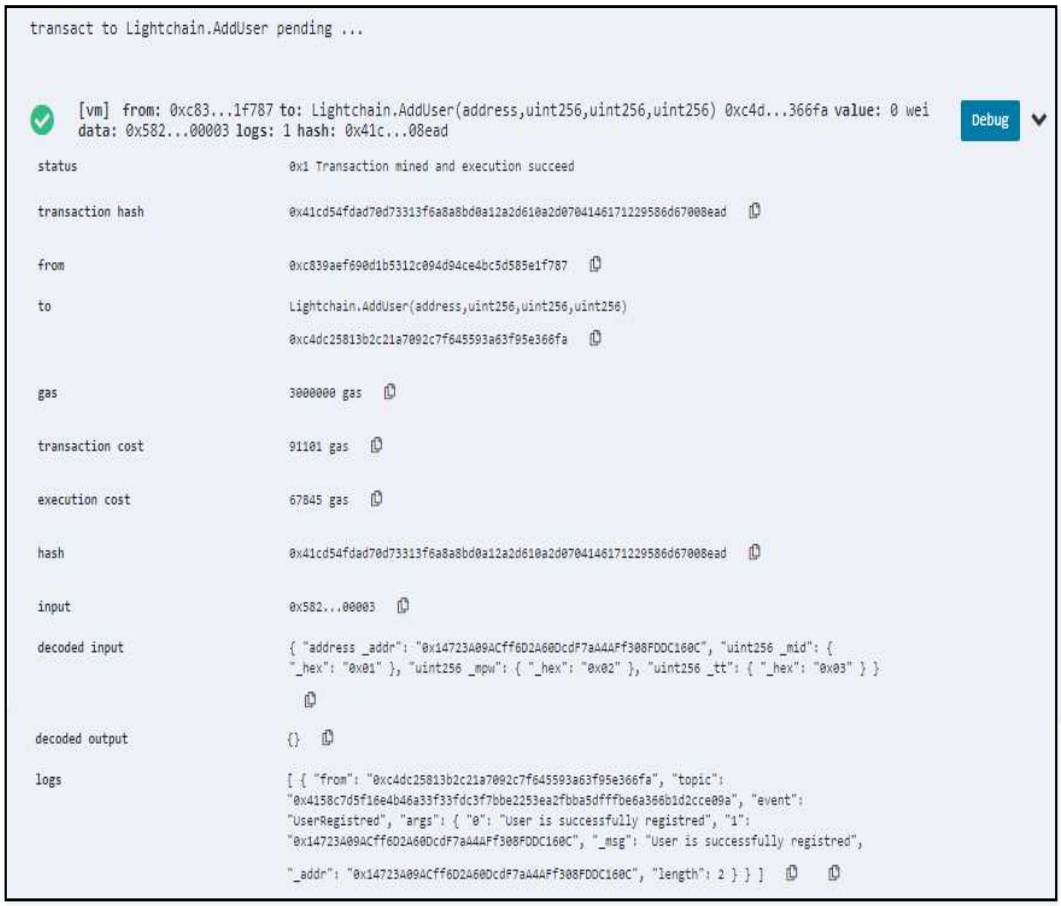


Figure 6.5: User registration transaction.

tries to send an authentication request, the operation fails with an error message, as shown in Figure 6.7.

### 6.5.2 Use case studies

Our proposed scheme provides lightweight and efficient remote user authentication and thus, it is suitable for major IoT applications including smart healthcare, smart industry and smart agriculture.

**Smart healthcare** : In remote healthcare monitoring system, medical sensors collect and send patients' physiological data periodically to cloud server. Medical professionals access the cloud data to diagnosis patients' diseases and monitor patients' health [192]. We consider that the medical profesional sends one authentication request per day. Thus, 30 transactions are triggered per month.

**Smart industry** : In smart manufacturing system, numerous automated machines are embedded with sensors and actuators to remotely control the production process

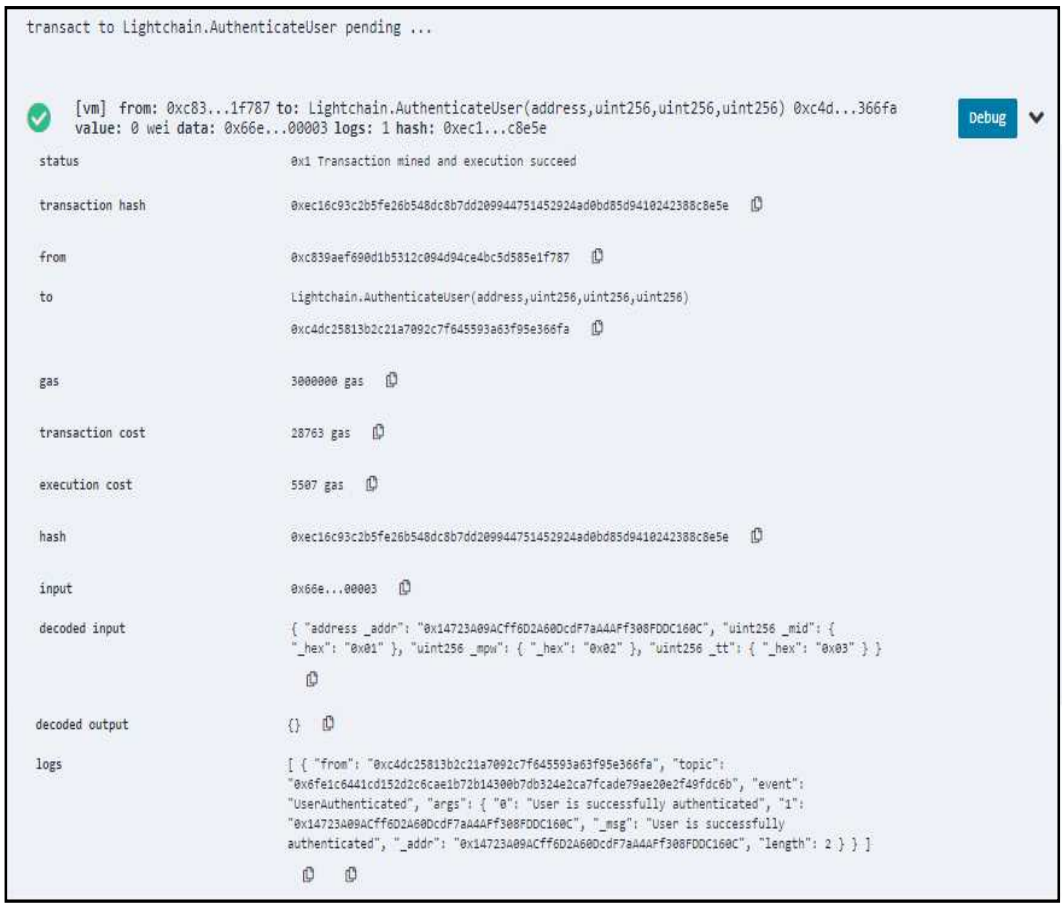


Figure 6.6: User authentication transaction.

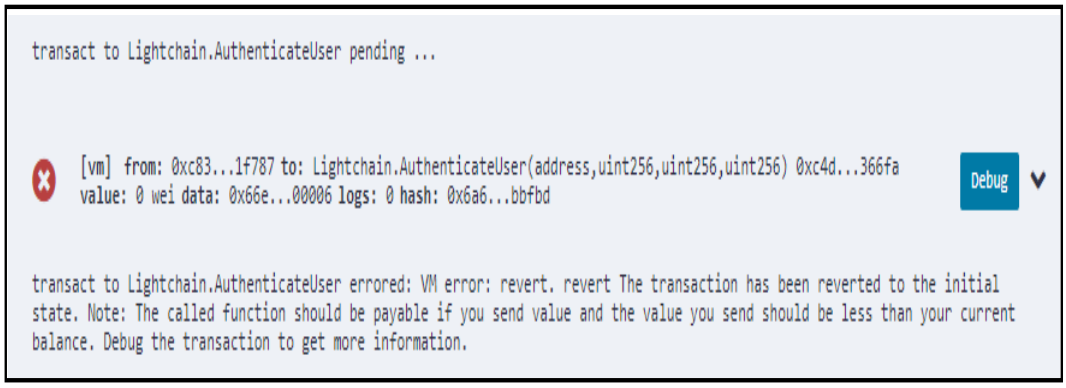


Figure 6.7: User authentication error.

Table 6.2: Execution cost of use cases studies.

| **Use case** | **Gas** | **ETH** |
|---|---|---|
| Smart healthcare | 165210 | 859.092 |
| Smart industry | 330420 | 1718.184 |
| Smart agriculture | 44056 | 229.0912 |

ETC: Ethereum cryptocurrency, 1 gas = 0.0052 ETH (23 August 2020)

and provide an efficient and reliable products [193]. Once the machine finishes the final product, it sends a message to factory cloud server. The remote user (owner) is considered to send two authentication request per day to control the fabrication process. Hence, 60 transactions are triggered per month.

**Smart agriculture** : In intelligent farming system or Greenhouse, sensors are deployed to measure different parameters such as temperature, humidity, irrigation, etc. These parameters are sent to the cloud for data processing and storage. Remote users access the cloud data in order to remotely control the greenhouse [194]. We consider that the remote user (farmer) sends two authentication requests per week. Consequently, 8 transactions are triggered per month.

As shown in Figure 6.6, the execution cost of transaction related to successful user authentication is $5507 gas$. Table 6.2 summarizes the financial cost of the considered use cases studies.

## 6.6 Comparative analysis

This section provides a comparative analysis of our proposed scheme and the schemes presented in [141–148] in terms of performance (*i.e.,* computational cost, communication cost and storage cost) and security requirements.

To compare the computation cost during user login and authentication phase, we denote $T_F$, $T_H$, $T_{SM}$, $T_{MAC}$, $T_S$, $T_A$ as time required by fuzzy extractor, hash function, ECC scalar multiplication, message authentication code, symmetric encryption/decryption, and asymmetric encryption/decryption, respectively. We do not consider the computation cost of XOR operation since it takes a negligible amount of time by comparison to other operations. According to [195], it is assumed that $T_F = T_{SM}$. To calculate the total execution time, we define $T_F = 2.226ms$, $T_H = 0.0023ms$,

Table 6.3: Comparison of computation cost of Lightchain scheme.

| **Scheme** | **Computational cost** | **Computational time(ms)** |
|---|---|---|
| Chaudhry et al. [141] | $13T_H$+$6T_{SM}$ | 13.3859 |
| Wazid et al. [142] | $1T_F$+$21T_H$+$8T_S$ | 2.3111 |
| Sharma et al. [143] | $16T_H$ | 0.0368 |
| Lin et al. [144] | $2T_{MAC}$+$6T_A$ | 23.1092 |
| Deebak et al. [145] | $1T_F$+$12T_H$+$6T_{SM}$+$2T_S$ | 15.6188 |
| Lee et al. [146] | $21T_H$+$2T_S$ | 0.0575 |
| Cui et al. [147] | $3T_H$+$4T_A$ | 15.4069 |
| Sadhukhan et al. [148] | $4T_H$+$6T_S$ | 0.0368 |
| Lightchain | $1T_F$+$19T_H$ | 2.2697 |

$T_{SM} = 2.226ms$, $T_{MAC} = 0.0046ms$, $T_S = 0.0046ms$ and $T_A = 3.85ms$ [121].

The computation cost required for our proposed scheme and related methods [141–148] is given in Table 6.3. We do not consider the initialization and user registration phases because they are executed only one-time, whereas the login and authentication phase is performed on the user's demand. Compared to [141, 142, 144, 145, 147], our proposed Lightchain is clearly more efficient in terms of computation cost because it is based on lightweight operations and requires less computation to authenticate remote users. However, it costs slightly more than the computation cost of the schemes presented in [143, 146, 148]. The small increased computation cost of our proposed scheme provides major security requirements including data privacy and session key agreement and resilience against known attacks such as guessing password and stolen smart card attacks that are not achieved by schemes [143, 146, 148].

To evaluate the communication cost, we assume that elliptic curve points are of 320 bits, the cipherdata size is equal to plaindata, the length of timestamp, identity, shared key, random number and output of hash function are of 160 bits.

Table 6.4 provides number of messages and bits required for remote user authentication. Our proposed scheme requires more communication cost compared to the related methods [141, 143, 144]. The reason is that in our Lightchain scheme, the remote user is authenticated through fog nodes before negotiating a session key with cloud server, while in [141, 143], the user authentication and session key agreement are only performed by cloud server. On the other hand, our proposed scheme requires less communication cost than the schemes [142, 146, 148] because short-length messages are transmitted in the network.

Table 6.4: Comparison of communication cost of Lightchain scheme.

| **Scheme** | **Number of messages** | **Communication cost(bits)** |
|---|---|---|
| Chaudhry et al. [141] | 3 | 1440 |
| Wazid et al. [142] | 4 | 3360 |
| Sharma et al. [143] | 3 | 1440 |
| Lin et al. [144] | 4 | 1280 |
| Deebak et al. [145] | 3 | 2880 |
| Lee et al. [146] | 4 | 3680 |
| Cui et al. [147] | 7 | 2880 |
| Sadhukhan et al. [148] | 4 | 3840 |
| Lightchain | 4 | 2880 |

Table 6.5: Comparison of storage cost of Lightchain scheme.

| **Scheme** | **Storage cost(bits)** |
|---|---|
| Chaudhry et al. [141] | 800 |
| Wazid et al. [142] | 1280 |
| Sharma et al. [143] | 640 |
| Lin et al. [144] | 640 |
| Deebak et al. [145] | 800 |
| Lee et al. [146] | 960 |
| Cui et al. [147] | - |
| Sadhukhan et al. [148] | 640 |
| Lightchain | 800 |

-: not mentioned

Table 6.5 presents the storage cost of user device related to login and authentication phase. Our proposed scheme requires 800 bits to store ($TT_i$, $D_i$, $E_i$, $\tau_i$) and $SK_i$. Compared to [142, 146], our scheme is more efficient in terms of storage cost. In contrast, the user device in Lightchain stores more bits of data in its memory than the schemes presented in [143, 144, 148]. This additional memory space provides more security against main user authentication attacks such as stolen smart device attack.

Figure 6.8 summarizes the performance comparison of our proposed scheme and related methods presented in [141–148].

Basically, a robust remote user authentication scheme must provide fundamental security requirements including resilience against known attacks (*e.g.,* user impersonation, insider attack, stolen user device, etc) and protection on different aspects (*e.g.,* mutual authentication, anonymity and privacy) [136]. In Table 6.6, we list the security requirements provided by our proposed scheme and other methods [141–148]. All the

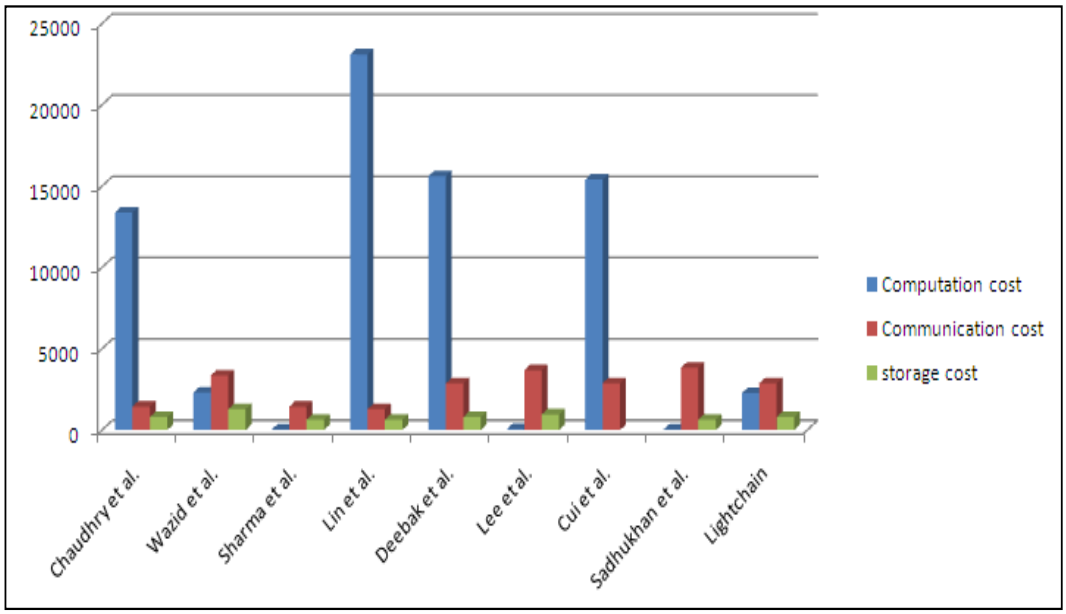


Figure 6.8: Performance comparison of Lightchain scheme.

schemes presented in [141–148] are prone to DoS attack because the user authentication process is based on a central entity. This allows attackers to exploit the authentication service provided by the central entity and thus, legal users are unable to authenticate to the system. The blockchain-based schemes [144, 147] are vulnerable to stolen user device attack and do not achieve data privacy, mutual authentication and session key agreement. By contrast, the Lightchain is more robust than other schemes [141–148], it resists main security attacks and provides important features such as mutual authentication, session key agreement, user anonymity and data privacy.

If we integrate both performance and security comparison results, our proposed scheme has similar costs as the schemes in [142, 143, 146, 148]. However, it is more lightweight than the schemes [141, 144, 145, 147]. In terms of security, our scheme is more secure than all related methods [141–148]. As a result, we claim that our proposed scheme is suitable for remote user authentication in different IoT applications due to its efficiency and robustness.

## 6.7 Conclusion

In this chapter, we proposed a lightweight blockchain-based scheme called Lightchain to authenticate remote user in IoT environments. We combined blockchain technology and fog computing and used cryptographic hash function to provide a distributed,

Table 6.6: Comparison of security requirements of Lightchain scheme.

| | [141] | [142] | [143] | [144] | [145] | [146] | [147] | [148] | Lightchain |
|---|---|---|---|---|---|---|---|---|---|
| R1 | × | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| R2 | × | ✓ | × | × | ✓ | ✓ | ✓ | ✓ | ✓ |
| R3 | × | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| R4 | × | ✓ | ✓ | × | ✓ | ✓ | × | ✓ | ✓ |
| R5 | × | ✓ | ✓ | × | ✓ | ✓ | × | × | ✓ |
| R6 | × | ✓ | ✓ | ✓ | ✓ | ✓ | × | × | ✓ |
| R7 | × | × | × | × | × | × | × | × | ✓ |
| R8 | ✓ | ✓ | ✓ | × | ✓ | ✓ | × | ✓ | ✓ |
| R9 | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | × | × | ✓ |
| R10 | ✓ | × | ✓ | × | ✓ | × | × | × | ✓ |

R1: user impersonation attack, R2: insider attack, R3: password guessing attack, R4: stolen user device attack, R5: replay attack, R6: MTM attack, R7: DoS attack, R8: mutual authentication and session key agreement, R9: user anonymity, R10: data privacy

scalable and efficient user authentication process. The formal security analysis using the widely-used AVISPA tool showed that our Lightchain is secure against active and passive attacks. Furthermore, we provided a comparative analysis with recent related methods in terms of computation cost, communication cost, storage cost and security requirements. The evaluation results demonstrated that our proposed scheme is more lightweight and robust.

# General conclusion

Nowadays, the IoT represents a major part of our daily life. Billions of intelligent and autonomous objects across the world are connected and communicate with each other. An IoT system uses wireless communications technologies to connect intelligent and autonomous objects. These objects have the ability to collect, analyze, process, generate and exchange information in order to provide advanced services.

The IoT is empowered by the proliferation of a myriad number of miniature sensors and actuators. Undeniably, WSNs are one of the main enabling technologies of the IoT since sensors can efficiently monitor critical environments in diverse domains.

With billions of new devices expected to be connected to the Internet in the next few years there is a wide variety of potential privacy and security risks faced by this expanding network. Hence, a secure architectures and frameworks are required which allow the IoT system to detect whether it is under attack, and therefore intensify the protection mechanisms. However, such system cannot use traditional security protocols (*e.g.* RSA, TLS) because they do not ensure good performance and are not suitable for resource-constrained IoT devices.

In this thesis, we explored the security and privacy issues and proposed lightweight and robust mechanisms to improve the IoT security without affecting the performance requirements. Initially, we analyzed the IoT security vulnerabilities and attacks of each layer. Then, we provided a taxonomy of IoT security attacks based on levels, purposes and countermeasures. We also presented a classification of IoT security requirements based on the attacks' purposes. This classification can help developers and researchers in designing new schemes to address security concerns of IoT systems. Subsequently, we proposed three security techniques to achieve fundamental requirements for IoT

systems. The common objective of these proposals is to provide a trade-off between efficiency and security (*i.e.* providing a good level of security with lightweight operations).

Firstly, we proposed a mutual authentication and session key agreement (MAKA) scheme based on ECC to secure communication in IoT-enabled WSNs. The informal security analysis showed that our proposed scheme is resilient to known security attacks such as replay, DoS, impersonation, etc. Moreover, we formally validated the MAKA scheme using the BAN logic and the widely-used AVISPA tool. In comparison with recent developed related methods, both the security and performance results showed that our proposed scheme is more secure and efficient for WSN-based IoT applications.

Secondly, we proposed a lightweight bio-inspired security (BOSS) scheme based on genetic algorithm and chaotic system. We adopted a symmetric encryption technique to encrypt the collected data before the data transmission process. Formal and informal security analyses showed that the proposed scheme resists active and passive attacks, and satisfies data privacy, integrity and authenticity with high confusion and diffusion. Simulation results using NS-3 showed that the proposed scheme is suitable for IoT with ressource-constrained devices. Compared to related encryption approaches, our proposed scheme is more efficient in terms of computation and storage costs. Furthermore, it is more secure and resilient to known IoT security attacks.

Thirdly, we proposed a lightweight blockchain-based scheme called Lightchain to authenticate remote user in IoT environments. We combined blockchain technology and fog computing and used cryptographic hash function to provide a distributed, scalable and efficient user authentication process. The formal security analysis using the widely-used AVISPA tool showed that our Lightchain is secure against active and passive attacks. Furthermore, we provided a comparative analysis with recent related methods in terms of computation cost, communication cost, storage cost and security requirements. The evaluation results demonstrated that our proposed scheme is more lightweight and robust.

## Future perspectives

Although the proposed security mechanisms for IoT are very promising and provide a good level of security with very acceptable performance, there are other important topics that we plan to carry out to strengthen the IoT security. Our future perspectives are summarized as follows:

- Improve the proposed schemes in terms of efficiency without affecting the security requirements.
- Enable end-user devices to securely search the encrypted data stored on the cloud server.
- Detect user anomaly in IoT environments using blockchain technology and machine learning algorithms.

# Bibliography


[1] Jayavardhana Gubbi, Rajkumar Buyya, Slaven Marusic, and Marimuthu Palaniswami. Internet of things (iot): A vision, architectural elements, and future directions. *Future generation computer systems*, 29(7):1645–1660, 2013.

[2] Ovidiu Vermesan and Peter Friess. *Internet of things: converging technologies for smart environments and integrated ecosystems*. River Publishers, 2013.

[3] Dave Evans. The internet of things: How the next evolution of the internet is changing everything. *CISCO white paper*, 1(2011):1–11, 2011.

[4] Ala Al-Fuqaha, Mohsen Guizani, Mehdi Mohammadi, Mohammed Aledhari, and Moussa Ayyash. Internet of things: A survey on enabling technologies, protocols, and applications. *IEEE Communications Surveys & Tutorials*, 17(4):2347–2376, 2015.

[5] Artur Marzano, David Alexander, Osvaldo Fonseca, Elverton Fazzion, Cristine Hoepers, Klaus Steding-Jessen, Marcelo HPC Chaves, Ítalo Cunha, Dorgival Guedes, and Wagner Meira. The evolution of bashlite and mirai iot botnets. In *2018 IEEE Symposium on Computers and Communications (ISCC)*, pages 00813–00818. IEEE, 2018.

[6] Ioannis Andrea, Chrysostomos Chrysostomou, and George Hadjichristofi. Internet of things: Security vulnerabilities and challenges. In *Computers and Communication (ISCC), 2015 IEEE Symposium on*, pages 180–187. IEEE, 2015.

[7] Sachchidanand Singh and Nirmala Singh. Internet of things (iot): Security challenges, business opportunities & reference architecture for e-commerce. In *Green Computing and Internet of Things (ICGCIoT), 2015 International Conference on*, pages 1577–1581. IEEE, 2015.

[8] Tuhin Borgohain, Uday Kumar, and Sugata Sanyal. Survey of security and privacy issues of internet of things. *arXiv preprint arXiv:1501.02211*, 2015.

[9] Zheng Yan, Peng Zhang, and Athanasios V Vasilakos. A survey on trust management for internet of things. *Journal of network and computer applications*, 42:120–134, 2014.

[10] Irfan Saif, Sean Peasley, and Arun Perinkolam. Safeguarding the internet of things: Being secure, vigilant, and resilient in the connected age. *Deloitte Review*, 17, 2015.

[11] Rodrigo Roman, Jianying Zhou, and Javier Lopez. On the features and challenges of security and privacy in distributed internet of things. *Computer Networks*, 57(10):2266–2279, 2013.

[12] Alessio Botta, Walter De Donato, Valerio Persico, and Antonio Pescapé. Integration of cloud computing and internet of things: a survey. *Future Generation Computer Systems*, 56:684–700, 2016.

[13] Carsten Bormann, Angelo P Castellani, and Zach Shelby. Coap: An application protocol for billions of tiny internet nodes. *IEEE Internet Computing*, 16(2):62–67, 2012.

[14] Sana Ullah, Murad Ali, Asdaque Hussain, and Kyung Sup Kwak. Applications of uwb technology. *arXiv preprint arXiv:0911.1681*, 2009.

[15] Kevin Curran, Amanda Millar, and Conor Mc Garvey. Near field communication. *International Journal of Electrical and Computer Engineering*, 2(3):371, 2012.

[16] Jie Lin, Wei Yu, Nan Zhang, Xinyu Yang, Hanlin Zhang, and Wei Zhao. A survey on internet of things: Architecture, enabling technologies, security and privacy, and applications. *IEEE Internet of Things Journal*, 4(5):1125–1142, 2017.

[17] Mustafa Kocakulak and Ismail Butun. An overview of wireless sensor networks towards internet of things. In *2017 IEEE 7th Annual Computing and Communication Workshop and Conference (CCWC)*, pages 1–6. IEEE, 2017.

[18] Xiaolin Jia, Quanyuan Feng, Taihua Fan, and Quanshui Lei. Rfid technology and its applications in internet of things (iot). In *2012 2nd international conference on consumer electronics, communications and networks (CECNet)*, pages 1282–1285. IEEE, 2012.

[19] ZigBee Specification. Zigbee document 053474r13. *ZgBee Standards Organization*, 2006.

[20] IEEE Working Group et al. Wireless medium access control and physical layer specifications for low-rate wireless personal area networks. *IEEE Standard*, 802(4):2003, 2003.

[21] Jianpo Li, Xuning Zhu, Ning Tang, and Jisheng Sui. Study on zigbee network architecture and routing algorithm. In *2010 2nd International Conference on Signal Processing Systems*, volume 2, pages V2–389. IEEE, 2010.

[22] SIG Bluetooth. Bluetooth core specification version 4.0. *Specification of the Bluetooth System*, 1:7, 2010.

[23] Gabriel Montenegro, Nandakishore Kushalnagar, Jonathan Hui, David Culler, et al. Transmission of ipv6 packets over ieee 802.15. 4 networks. *Internet proposed standard RFC*, 4944:130, 2007.

[24] Geoff Mulligan. The 6lowpan architecture. In *Proceedings of the 4th workshop on Embedded networked sensors*, pages 78–82, 2007.

[25] Tim Winter, Pascal Thubert, Anders Brandt, Jonathan W Hui, Richard Kelsey, Philip Levis, Kris Pister, Rene Struik, Jean-Philippe Vasseur, Roger K Alexander, et al. Rpl: Ipv6 routing protocol for low-power and lossy networks. *rfc*, 6550:1–157, 2012.

[26] LoRa Alliance. Lorawan 1.1 specification. *technical specification*, 2017.

[27] Zach Shelby, Klaus Hartke, and Carsten Bormann. The constrained application protocol (coap). *rfc*, 7252, 2014.

[28] Qi Jing, Athanasios V Vasilakos, Jiafu Wan, Jingwei Lu, and Dechao Qiu. Security of the internet of things: perspectives and challenges. *Wireless Networks*, 20(8):2481–2501, 2014.

[29] Silvio Cesare. Breaking the security of physical devices. *Presentation at Blackhat*, 14, 2014.

[30] Jorge Granjal, Edmundo Monteiro, and Jorge Sá Silva. Security for the internet of things: a survey of existing protocols and open research issues. *IEEE Communications Surveys & Tutorials*, 17(3):1294–1312, 2015.

[31] Sabrina Sicari, Alessandra Rizzardi, Luigi Alfredo Grieco, and Alberto Coen-Porisini. Security, privacy and trust in internet of things: The road ahead. *Computer networks*, 76:146–164, 2015.

[32] Kai Zhao and Lina Ge. A survey on the internet of things security. In *Computational Intelligence and Security (CIS), 2013 9th International Conference on*, pages 663–667. IEEE, 2013.

[33] M Vivekananda Bharathi, Rama Chaithanya Tanguturi, C Jayakumar, and K Selvamani. Node capture attack in wireless sensor network: A survey. In *2012 IEEE International Conference on Computational Intelligence and Computing Research*, pages 1–3. IEEE, 2012.

[34] Shoukat Ali, Muazzam A Khan, Jawad Ahmad, Asad W Malik, and Anis ur Rehman. Detection and prevention of black hole attacks in iot & wsn. In *2018 Third International Conference on Fog and Mobile Edge Computing (FMEC)*, pages 217–226. IEEE, 2018.

[35] Vinay Soni, Pratik Modi, and Vishvash Chaudhri. Detecting sinkhole attack in wireless sensor network. *International Journal of Application or Innovation in Engineering & Management*, 2(2):29–32, 2013.

[36] Phillip Lee, Andrew Clark, Linda Bushnell, and Radha Poovendran. A passivity framework for modeling and mitigating wormhole attacks on networked control systems. *IEEE Transactions on Automatic Control*, 59(12):3224–3237, 2014.

[37] Kuan Zhang, Xiaohui Liang, Rongxing Lu, and Xuemin Shen. Sybil attacks and their defenses in the internet of things. *IEEE Internet of Things Journal*, 1(5):372–383, 2014.

[38] Benjamin Khoo. Rfid as an enabler of the internet of things: Issues of security and privacy. In *Internet of Things (iThings/CPSCom), 2011 International Conference on and 4th International Conference on Cyber, Physical and Social Computing*, pages 709–712. IEEE, 2011.

[39] Mukrimah Nawir, Amiza Amir, Naimah Yaakob, and Ong Bi Lynn. Internet of things (iot): Taxonomy of security attacks. In *Electronic Design (ICED), 2016 3rd International Conference on*, pages 321–326. IEEE, 2016.

[40] Ebraheim Alsaadi and Abdallah Tubaishat. Internet of things: features, challenges, and vulnerabilities. *International Journal of Advanced Computer Science and Information Technology*, 4(1):1–13, 2015.

[41] Aristides Mpitziopoulos, Damianos Gavalas, Charalampos Konstantopoulos, and Grammati Pantziou. A survey on jamming attacks and countermeasures in wsns. *IEEE Communications Surveys & Tutorials*, 11(4), 2009.

[42] Tom N Jagatic, Nathaniel A Johnson, Markus Jakobsson, and Filippo Menczer. Social phishing. *Communications of the ACM*, 50(10):94–100, 2007.

[43] Yasmine Harbi, Zibouda Aliouat, Saad Harous, Abdelhak Bentaleb, and Allaoua Refoufi. A review of security in internet of things. *Wireless Personal Communications*, 108(1):325–344, 2019.

[44] T Zillner and F Eichelberger. Zigbee smart homes: A hackers open house. In *Proc. CRESTCon Conf.*, 2016.

[45] Xianghui Cao, Devu Manikantan Shila, Yu Cheng, Zequ Yang, Yang Zhou, and Jiming Chen. Ghost-in-zigbee: Energy depletion attack on zigbee-based wireless networks. *IEEE Internet of Things Journal*, 3(5):816–829, 2016.

[46] Luigi Coppolino, Valerio DAlessandro, Salvatore DAntonio, Leonid Levy, and Luigi Romano. My smart home is under attack. In *2015 IEEE 18th International Conference on Computational Science and Engineering*, pages 145–151. IEEE, 2015.

[47] Philipp Morgner, Stephan Mattejat, Zinaida Benenson, Christian Müller, and Frederik Armknecht. Insecure to the touch: attacking zigbee 3.0 via touchlink commissioning. In *Proceedings of the 10th ACM Conference on Security and Privacy in Wireless and Mobile Networks*, pages 230–240, 2017.

[48] Mike Ryan. Bluetooth: With low energy comes low security. In *7th {USENIX} Workshop on Offensive Technologies ({WOOT} 13)*, 2013.

[49] Andrew Y Lindell. Attacks on the pairing protocol of bluetooth v2. 1. *Black Hat USA, Las Vegas, Nevada*, 2008.

[50] Wondimu K Zegeye. Exploiting bluetooth low energy pairing vulnerability in telemedicine. International Foundation for Telemetering, 2015.

[51] Tomas Rosa. Bypassing passkey authentication in bluetooth low energy. *IACR Cryptol. ePrint Arch.*, 2013:309, 2013.

[52] Mengmei Ye, Nan Jiang, Hao Yang, and Qiben Yan. Security analysis of internet-of-things: A case study of august smart lock. In *2017 IEEE conference on computer communications workshops (INFOCOM WKSHPS)*, pages 499–504. IEEE, 2017.

[53] René Hummen, Jens Hiller, Hanno Wirtz, Martin Henze, Hossein Shafagh, and Klaus Wehrle. 6lowpan fragmentation attacks and mitigation mechanisms. In *Proceedings of the sixth ACM conference on Security and privacy in wireless and mobile networks*, pages 55–66, 2013.

[54] A Rghiout, A Khannous, and M Bouhorma. Denial-of-service attacks on 6lowpan-rpl networks: Issues and practical solutions. *Journal of Advanced Computer Science & Technology*, 3(2):143–153, 2014.

[55] Pavan Pongle and Gurunath Chavan. A survey: Attacks on rpl and 6lowpan in iot. In *2015 International conference on pervasive computing (ICPC)*, pages 1–6. IEEE, 2015.

[56] Anthea Mayzaud, Remi Badonnel, and Isabelle Chrisment. A taxonomy of attacks in rpl-based internet of things. *International Journal of Network Security*, 18(3):459–473, 2016.

[57] Robert Miller. Lora security: Building a secure lora solution. *MWR Labs Whitepaper*, 2016.

[58] Xueying Yang, Evgenios Karampatzakis, Christian Doerr, and Fernando Kuipers. Security vulnerabilities in lorawan. In *2018 IEEE/ACM Third International Conference on Internet-of-Things Design and Implementation (IoTDI)*, pages 129–140. IEEE, 2018.

[59] Reem Abdul Rahman and Babar Shah. Security analysis of iot protocols: A focus in coap. In *2016 3rd MEC international conference on big data and smart city (ICBDSC)*, pages 1–7. IEEE, 2016.

[60] J Cynthia, H Parveen Sultana, MN Saroja, and J Senthil. Security protocols for iot. In *Ubiquitous computing and computing security of IoT*, pages 1–28. Springer, 2019.

[61] Meena Singh, MA Rajan, VL Shivraj, and P Balamuralidhar. Secure mqtt for internet of things (iot). In *2015 Fifth International Conference on Communication Systems and Network Technologies*, pages 746–751. IEEE, 2015.

[62] Md Mahmud Hossain, Maziar Fotouhi, and Ragib Hasan. Towards an analysis of security issues, challenges, and open problems in the internet of things. In *Services (SERVICES), 2015 IEEE World Congress on*, pages 21–28. IEEE, 2015.

[63] Diego M Mendez, Ioannis Papapanagiotou, and Baijian Yang. Internet of things: Survey on security and privacy. *arXiv preprint arXiv:1707.01879*, 2017.

[64] Daniele Miorandi, Sabrina Sicari, Francesco De Pellegrini, and Imrich Chlamtac. Internet of things: Vision, applications and research challenges. *Ad hoc networks*, 10(7):1497–1516, 2012.

[65] Sarfraz Alam, Mohammad MR Chowdhury, and Josef Noll. Interoperability of security-enabled internet of things. *Wireless Personal Communications*, 61(3):567–586, 2011.

[66] Sachin Babar, Antonietta Stango, Neeli Prasad, Jaydip Sen, and Ramjee Prasad. Proposed embedded security framework for internet of things (iot). In *Wireless Communication, Vehicular Technology, Information Theory and Aerospace & Electronic Systems Technology (Wireless VITAE), 2011 2nd International Conference on*, pages 1–5. IEEE, 2011.

[67] Rolf H Weber. Internet of things: Privacy issues revisited. *Computer Law & Security Review*, 31(5):618–627, 2015.

[68] Sridipta Misra, Muthucumaru Maheswaran, and Salman Hashmi. *Security challenges and approaches in internet of things*. Springer, 2017.

[69] Arbia Riahi Sfar, Enrico Natalizio, Yacine Challal, and Zied Chtourou. A roadmap for security challenges in the internet of things. *Digital Communications and Networks*, 2017.

[70] Luigi Atzori, Antonio Iera, and Giacomo Morabito. The internet of things: A survey. *Computer networks*, 54(15):2787–2805, 2010.

[71] Andrey Bogdanov, Miroslav Knežević, Gregor Leander, Deniz Toz, Kerem Varıcı, and Ingrid Verbauwhede. Spongent: A lightweight hash function. In *International*

*Workshop on Cryptographic Hardware and Embedded Systems*, pages 312–325. Springer, 2011.

[72] Jean-Philippe Aumasson, Luca Henzen, Willi Meier, and María Naya-Plasencia. Quark: A lightweight hash. *Journal of cryptology*, 26(2):313–339, 2013.

[73] Mohammad Reza Sohizadeh Abyaneh. Security analysis of lightweight schemes for rfid systems. *Ph.D. thesis, The University of Bergen, Norway*, 2012.

[74] Rachel Greenstadt and Jacob Beal. Cognitive security for personal devices. In *Proceedings of the 1st ACM workshop on Workshop on AISec*, pages 27–30. ACM, 2008.

[75] Jing Liu, Yang Xiao, and CL Philip Chen. Internet of things' authentication and access control. *International Journal of Security and Networks*, 7(4):228–241, 2012.

[76] Al-Sakib Khan Pathan, Hyung-Woo Lee, and Choong Seon Hong. Security in wireless sensor networks: issues and challenges. In *Advanced Communication Technology, 2006. ICACT 2006. The 8th International Conference*, volume 2, pages 6–pp. IEEE, 2006.

[77] Edewede Oriwoh, Haider al Khateeb, and Marc Conrad. Responsibility and non-repudiation in resource-constrained internet of things scenarios. International Conference on Computing and Technology Innovation (CTI 2015), 2016.

[78] Louis Coetzee and Johan Eksteen. The internet of things-promise for the future? an introduction. In *IST-Africa Conference Proceedings, 2011*, pages 1–9. IEEE, 2011.

[79] Sandro Etalle, Jerry den Hartog, and Stephen Marsh. Trust and punishment. In *Proceedings of the 1st international conference on Autonomic computing and communication systems*, page 5. ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), 2007.

[80] Prabhakaran Kasinathan, Claudio Pastrone, Maurizio A Spirito, and Mark Vinkovits. Denial-of-service detection in 6lowpan based internet of things. In *Wireless and Mobile Computing, Networking and Communications (WiMob), 2013 IEEE 9th International Conference on*, pages 600–607. IEEE, 2013.

[81] Hui Suo, Jiafu Wan, Caifeng Zou, and Jianqi Liu. Security in the internet of things: a review. In *Computer Science and Electronics Engineering (ICCSEE), 2012 international conference on*, volume 3, pages 648–651. IEEE, 2012.

[82] Vipindev Adat and BB Gupta. Security in internet of things: issues, challenges, taxonomy, and architecture. *Telecommunication Systems*, 67(3):423–441, 2017.

[83] Nataliia Neshenko, Elias Bou-Harb, Jorge Crichigno, Georges Kaddoum, and Nasir Ghani. Demystifying iot security: an exhaustive survey on iot vulnerabilities and a first empirical look on internet-scale iot exploitations. *IEEE Communications Surveys & Tutorials*, 21(3):2702–2733, 2019.

[84] Flavio Bonomi, Rodolfo Milito, Jiang Zhu, and Sateesh Addepalli. Fog computing and its role in the internet of things characterization of fog computing. In *Proc. MCC*, pages 13–17, 2016.

[85] Arwa Alrawais, Abdulrahman Alhothaily, Chunqiang Hu, Xiaoshuang Xing, and Xiuzhen Cheng. An attribute-based encryption scheme to secure fog communications. *IEEE access*, 5:9131–9138, 2017.

[86] Pengfei Hu, Huansheng Ning, Tie Qiu, Houbing Song, Yanna Wang, and Xuanxia Yao. Security and privacy preservation scheme of face identification and resolution framework using fog computing in internet of things. *IEEE Internet of Things Journal*, 4(5):1143–1155, 2017.

[87] Prosanta Gope. Laap: Lightweight anonymous authentication protocol for d2d-aided fog computing paradigm. *computers & security*, 86:223–237, 2019.

[88] Rongxing Lu, Kevin Heung, Arash Habibi Lashkari, and Ali A Ghorbani. A lightweight privacy-preserving data aggregation scheme for fog computing-enhanced iot. *IEEE Access*, 5:3302–3312, 2017.

[89] Xue Yang, Fan Yin, and Xiaohu Tang. A fine-grained and privacy-preserving query scheme for fog computing-enhanced location-based service. *Sensors*, 17(7):1611, 2017.

[90] Zhitao Guan, Yue Zhang, Longfei Wu, Jun Wu, Jing Li, Yinglong Ma, and Jingjing Hu. Appa: An anonymous and privacy preserving data aggregation scheme for fog-enhanced iot. *Journal of Network and Computer Applications*, 125:82–92, 2019.

[91] Kwasi Boakye-Boateng, Eric Kuada, Emmanuel Antwi-Boasiako, and Emmanuel Djaba. Encryption protocol for resource-constrained devices in fog-based iot using one-time pads. *IEEE Internet of Things Journal*, 6(2):3925–3933, 2019.

[92] Pedro H Vilela, Joel JPC Rodrigues, Petar Solic, Kashif Saleem, and Vasco Furtado. Performance evaluation of a fog-assisted iot solution for e-health applications. *Future Generation Computer Systems*, 97:379–386, 2019.

[93] Hui Li and Tao Jing. A ciphertext-policy attribute-based encryption scheme with public verification for an iot-fog-cloud architecture. *Procedia Computer Science*, 174:243–251, 2020.

[94] Djamel Eddine Kouicem, Abdelmadjid Bouabdallah, and Hicham Lakhlef. Internet of things security: A top-down survey. *Computer Networks*, 141:199–221, 2018.

[95] Kubra Kalkan and Sherali Zeadally. Securing internet of things with software defined networking. *IEEE Communications Magazine*, 56(9):186–192, 2017.

[96] Xiaoliang Wang, Ke Xu, Wenlong Chen, Qi Li, Meng Shen, and Bo Wu. Id-based sdn for the internet of things. *IEEE Network*, 34(4):76–83, 2020.

[97] Ola Salman, Sarah Abdallah, Imad H Elhajj, Ali Chehab, and Ayman Kayssi. Identity-based authentication scheme for the internet of things. In *2016 IEEE Symposium on Computers and Communication (ISCC)*, pages 1109–1111. IEEE, 2016.

[98] Junxia Li, Jinjin Cai, Fazlullah Khan, Ateeq Ur Rehman, Venki Balasubramaniam, Jiangfeng Sun, and P Venu. A secured framework for sdn-based edge computing in iot-enabled healthcare system. *IEEE Access*, 8:135479–135490, 2020.

[99] Dongdong Ma and Yijie Shi. A lightweight encryption algorithm for edge networks in software-defined industrial internet of things. In *2019 IEEE 5th International Conference on Computer and Communications (ICCC)*, pages 1489–1493. IEEE, 2019.

[100] Peter Bull, Ron Austin, Evgenii Popov, Mak Sharma, and Richard Watson. Flow based security for iot devices using an sdn gateway. In *2016 IEEE 4th*

*international conference on future internet of things and cloud (FiCloud)*, pages 157–163. IEEE, 2016.

[101] Suman Sankar Bhunia and Mohan Gurusamy. Dynamic attack detection and mitigation in iot using sdn. In *2017 27th International telecommunication networks and applications conference (ITNAC)*, pages 1–6. IEEE, 2017.

[102] Anis Herbadji, Hadjer Goumidi, Yasmine Harbi, Khadidja Medani, and Zibouda Aliouat. Blockchain for internet of vehicles security. *Blockchain for Cybersecurity and Privacy: Architectures, Challenges, and Applications*, page 159, 2020.

[103] Axel Moinet, Benoît Darties, and Jean-Luc Baril. Blockchain based trust & authentication for decentralized sensor networks. *arXiv preprint arXiv:1706.01730*, 2017.

[104] Mohamed Tahar Hammi, Badis Hammi, Patrick Bellot, and Ahmed Serhrouchni. Bubbles of trust: A decentralized blockchain-based authentication system for iot. *Computers & Security*, 78:126–142, 2018.

[105] Chao Lin, Debiao He, Neeraj Kumar, Xinyi Huang, Pandi Vijayakumar, and Kim-Kwang Raymond Choo. Homechain: a blockchain-based secure mutual authentication system for smart homes. *IEEE Internet of Things Journal*, 7(2):818–829, 2019.

[106] Sunghyuck Hong. P2p networking based internet of things (iot) sensor node authentication by blockchain. *Peer-to-Peer Networking and Applications*, 13(2):579–589, 2020.

[107] Umair Khalid, Muhammad Asim, Thar Baker, Patrick CK Hung, Muhammad Adnan Tariq, and Laura Rafferty. A decentralized lightweight blockchain-based authentication mechanism for iot systems. *Cluster Computing*, pages 1–21, 2020.

[108] Asaph Azaria, Ariel Ekblaw, Thiago Vieira, and Andrew Lippman. Medrec: Using blockchain for medical data access and permission management. In *2016 2nd International Conference on Open and Big Data (OBD)*, pages 25–30. IEEE, 2016.

[109] Aafaf Ouaddah, Anas Abou Elkalam, and Abdellah Ait Ouahman. Fairaccess: a new blockchain-based access control framework for the internet of things. *Security and Communication Networks*, 9(18):5943–5964, 2016.

[110] Ali Dorri, Salil S Kanhere, Raja Jurdak, and Praveen Gauravaram. Blockchain for iot security and privacy: The case study of a smart home. In *2017 IEEE international conference on pervasive computing and communications workshops (PerCom workshops)*, pages 618–623. IEEE, 2017.

[111] Jun Lin, Zhiqi Shen, and Chunyan Miao. Using blockchain technology to build trust in sharing lorawan iot. In *Proceedings of the 2nd International Conference on Crowd Science and Engineering*, pages 38–43, 2017.

[112] Roberto Di Pietro, Xavier Salleras, Matteo Signorini, and Erez Waisbard. A blockchain-based trust system for the internet of things. In *Proceedings of the 23nd ACM on Symposium on Access Control Models and Technologies*, pages 77–83, 2018.

[113] Volkan Dedeoglu, Raja Jurdak, Guntur D Putra, Ali Dorri, and Salil S Kanhere. A trust architecture for blockchain in iot. In *Proceedings of the 16th EAI International Conference on Mobile and Ubiquitous Systems: Computing, Networking and Services*, pages 190–199, 2019.

[114] Bo Tang, Hongjuan Kang, Jingwen Fan, Qi Li, and Ravi Sandhu. Iot passport: A blockchain-based trust framework for collaborative internet-of-things. In *Proceedings of the 24th ACM symposium on access control models and technologies*, pages 83–92, 2019.

[115] Yongping Zhang, Xiwei Xu, Ang Liu, Qinghua Lu, Lida Xu, and Fei Tao. Blockchain-based trust mechanism for iot-based smart manufacturing system. *IEEE Transactions on Computational Social Systems*, 6(6):1386–1394, 2019.

[116] Sumit Singh Dhanda, Brahmjit Singh, and Poonam Jindal. Lightweight cryptography: A solution to secure iot. *Wireless Personal Communications*, pages 1–34, 2020.

[117] Muhammad Usman, Irfan Ahmed, M Imran Aslam, Shujaat Khan, and Usman Ali Shah. Sit: a lightweight encryption algorithm for secure internet of things. *arXiv preprint arXiv:1704.08688*, 2017.

[118] Romana Shahzadi, Syed Muhammad Anwar, Farhan Qamar, Mudassar Ali, and Joel JPC Rodrigues. Chaos based enhanced rc5 algorithm for security and integrity of clinical images in remote health monitoring. *IEEE Access*, 7:52858–52870, 2019.

[119] Masoumeh Sharafi, Faranak Fotouhi-Ghazvini, Mohsen Shirali, and Mona Ghassemian. A low power cryptography solution based on chaos theory in wireless sensor nodes. *IEEE Access*, 7:8737–8753, 2019.

[120] Hassan Noura, Raphaël Couturier, Congduc Pham, and Ali Chehab. Lightweight stream cipher scheme for resource-constrained iot devices. In *2019 International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob)*, pages 1–8. IEEE, 2019.

[121] Yasmine Harbi, Zibouda Aliouat, Allaoua Refoufi, Saad Harous, and Abdelhak Bentaleb. Enhanced authentication and key management scheme for securing data transmission in the internet of things. *Ad Hoc Networks*, 94:101948, 2019.

[122] Goiuri Peralta, Raul G Cid-Fuentes, Josu Bilbao, and Pedro M Crespo. Homomorphic encryption and network coding in iot architectures: Advantages and future challenges. *Electronics*, 8(8):827, 2019.

[123] Umasankararao Varri, Syamkumar Pasupuleti, and KV Kadambari. A scoping review of searchable encryption schemes in cloud computing: taxonomy, methods, and recent developments. *The Journal of Supercomputing*, 76(4):3013–3042, 2020.

[124] Jaweher Zouari, Mohamed Hamdi, and Tai-Hoon Kim. A privacy-preserving homomorphic encryption scheme for the internet of things. In *2017 13th International Wireless Communications and Mobile Computing Conference (IWCMC)*, pages 1939–1944. IEEE, 2017.

[125] Shuai Li, Miao Li, Haitao Xu, and Xianwei Zhou. Searchable encryption scheme for personalized privacy in iot-based big data. *Sensors*, 19(5):1059, 2019.

[126] Fatima Hussain, Rasheed Hussain, Syed Ali Hassan, and Ekram Hossain. Machine learning in iot security: current solutions and future challenges. *IEEE Communications Surveys & Tutorials*, 2020.

[127] Mohammed Ali Al-Garadi, Amr Mohamed, Abdulla Al-Ali, Xiaojiang Du, Ihsan Ali, and Mohsen Guizani. A survey of machine and deep learning methods for

internet of things (iot) security. *IEEE Communications Surveys & Tutorials*, 2020.

[128] Muhamed Turkanović, Boštjan Brumen, and Marko 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*, 20:96–112, 2014.

[129] Mohammad Sabzinejad Farash, Muhamed Turkanović, Saru Kumari, and Marko Hölbl. An efficient user authentication and key agreement scheme for heterogeneous wireless sensor network tailored for the internet of things environment. *Ad Hoc Networks*, 36:152–176, 2016.

[130] Jian Shen, Shaohua Chang, Jun Shen, Qi Liu, and Xingming Sun. A lightweight multi-layer authentication protocol for wireless body area networks. *Future Generation Computer Systems*, 78:956–963, 2018.

[131] Fan Wu, Lili Xu, Saru Kumari, and Xiong Li. A privacy-preserving and provable user authentication scheme for wireless sensor networks based on internet of things security. *Journal of Ambient Intelligence and Humanized Computing*, 8(1):101–116, 2017.

[132] Dheerendra Mishra, P Vijayakumar, Venkatasamy Sureshkumar, Ruhul Amin, SK Hafizul Islam, and Prosanta Gope. Efficient authentication protocol for secure multimedia communications in iot-enabled wireless sensor networks. *Multimedia Tools and Applications*, 77(14):18295–18325, 2018.

[133] Chenyu Wang, Guoai Xu, and Jing Sun. An enhanced three-factor user authentication scheme using elliptic curve cryptosystem for wireless sensor networks. *Sensors*, 17(12):2946, 2017.

[134] Xiong Li, Jianwei Niu, Md Zakirul Alam Bhuiyan, Fan Wu, Marimuthu Karuppiah, and Saru Kumari. A robust ecc-based provable secure authentication protocol with privacy preserving for industrial internet of things. *IEEE Transactions on Industrial Informatics*, 14(8):3599–3609, 2018.

[135] Amjad Mehmood, Muhammad Muneer Umar, and Houbing Song. Icmds: Secure inter-cluster multiple-key distribution scheme for wireless sensor networks. *Ad Hoc Networks*, 55:97–106, 2017.

[136] Ruhul Amin, SK Hafizul Islam, GP Biswas, and Mohammad S Obaidat. A robust mutual authentication protocol for wsn with multiple base-stations. *Ad Hoc Networks*, 75:1–18, 2018.

[137] Ankur Gupta, Meenakshi Tripathi, Tabish Jamil Shaikh, and Aakar Sharma. A lightweight anonymous user authentication and key establishment scheme for wearable devices. *Computer Networks*, 149:29–42, 2019.

[138] Tianyi Song, Ruinian Li, Bo Mei, Jiguo Yu, Xiaoshuang Xing, and Xiuzhen Cheng. A privacy preserving communication protocol for iot applications in smart homes. *IEEE Internet of Things Journal*, 4(6):1844–1852, 2017.

[139] Shadi Aljawarneh, Muneer Bani Yassein, et al. A resource-efficient encryption algorithm for multimedia big data. *Multimedia Tools and Applications*, 76(21):22703–22724, 2017.

[140] Mohamed Elhoseny, Gustavo Ramírez-González, Osama M Abu-Elnasr, Shihab A Shawkat, N Arunkumar, and Ahmed Farouk. Secure medical data transmission model for iot-based healthcare systems. *Ieee Access*, 6:20596–20608, 2018.

[141] Shehzad Ashraf Chaudhry, Husnain Naqvi, Khalid Mahmood, Hafiz Farooq Ahmad, and Muhammad Khurram Khan. An improved remote user authentication scheme using elliptic curve cryptography. *Wireless Personal Communications*, 96(4):5355–5373, 2016.

[142] Mohammad Wazid, Ashok Kumar Das, Vanga Odelu, Neeraj Kumar, Mauro Conti, and Minho Jo. Design of secure user authenticated key management protocol for generic iot networks. *IEEE Internet of Things Journal*, 5(1):269–282, 2017.

[143] Geeta Sharma and Sheetal Kalra. A lightweight multi-factor secure smart card based remote user authentication scheme for cloud-iot applications. *Journal of information security and applications*, 42:95–106, 2018.

[144] Chao Lin, Debiao He, Neeraj Kumar, Xinyi Huang, Pandi Vijayakumar, and Kim-Kwang Raymond Choo. Homechain: a blockchain-based secure mutual authentication system for smart homes. *IEEE Internet of Things Journal*, 7(2):818–829, 2019.

[145] Bakkiam David Deebak, Fadi Al-Turjman, Moayad Aloqaily, and Omar Alfandi. An authentic-based privacy preservation protocol for smart e-healthcare systems in iot. *IEEE Access*, 7:135632–135649, 2019.

[146] Hakjun Lee, Dongwoo Kang, Jihyeon Ryu, Dongho Won, Hyoungshick Kim, and Youngsook Lee. A three-factor anonymous user authentication scheme for internet of things environments. *Journal of Information Security and Applications*, 52:102494, 2020.

[147] Zhihua Cui, XUE Fei, Shiqiang Zhang, Xingjuan Cai, Yang Cao, Wensheng Zhang, and Jinjun Chen. A hybrid blockchain-based identity authentication scheme for multi-wsn. *IEEE Transactions on Services Computing*, 13(2):241–251, 2020.

[148] Dipanwita Sadhukhan, Sangram Ray, GP Biswas, MK Khan, and Mou Dasgupta. A lightweight remote user authentication scheme for iot communication using elliptic curve cryptography. *Journal of supercomputing*, 2020.

[149] Andrew Whitmore, Anurag Agarwal, and Li Da Xu. The internet of thingsa survey of topics and trends. *Information Systems Frontiers*, 17(2):261–274, 2015.

[150] Yasmine Harbi, Zibouda Aliouat, and Sarra Hammoudi. Enhancement of iot applications dependability using bayesian networks. In *Computational Intelligence and Its Applications: 6th IFIP TC 5 International Conference, CIIA 2018, Oran, Algeria, May 8-10, 2018, Proceedings 6*, pages 487–497. Springer, 2018.

[151] Zhengguo Sheng, Chinmaya Mahapatra, Chunsheng Zhu, and Victor Leung. Recent advances in industrial wireless sensor networks towards efficient management in iot. *IEEE access*, 3:622–637, 2015.

[152] Luca Catarinucci, Danilo De Donno, Luca Mainetti, Luca Palano, Luigi Patrono, Maria Laura Stefanizzi, and Luciano Tarricone. An iot-aware architecture for smart healthcare systems. *IEEE Internet of Things Journal*, 2(6):515–526, 2015.

[153] Debiao He and Sherali Zeadally. Authentication protocol for an ambient assisted living system. *IEEE Communications Magazine*, 53(1):71–77, 2015.

[154] Ian F Akyildiz, Weilian Su, Yogesh Sankarasubramaniam, and Erdal Cayirci. Wireless sensor networks: a survey. *Computer networks*, 38(4):393–422, 2002.

[155] Yasmine Harbi, Zibouda Aliouat, Saad Harous, and Abdelhak Bentaleb. Secure data transmission scheme based on elliptic curve cryptography for internet of things. In *International Symposium on Modelling and Implementation of Complex Systems*, pages 34–46. Springer, 2018.

[156] Jangirala Srinivas, Sourav Mukhopadhyay, and Dheerendra Mishra. Secure and efficient user authentication scheme for multi-gateway wireless sensor networks. *Ad Hoc Networks*, 54:147–169, 2017.

[157] Darrel Hankerson, Alfred J Menezes, and Scott Vanstone. *Guide to elliptic curve cryptography*. Springer Science & Business Media, 2006.

[158] Joseph H Silverman. *The arithmetic of elliptic curves*, volume 106. Springer Science & Business Media, 2009.

[159] Jeffrey Hoffstein, Jill Catherine Pipher, Joseph H Silverman, and Joseph H Silverman. *An introduction to mathematical cryptography*, volume 1. Springer, 2008.

[160] Alfred Menezes. An introduction to pairing-based cryptography. *Recent trends in cryptography*, 477:47–65, 2009.

[161] Lina Aliouat and Zibouda Aliouat. Improved wsn life time duration through adaptive clustering, duty cycling and sink mobility. In *Proceedings of the 2016 8th International Conference on Information Management and Engineering*, pages 36–40. ACM, 2016.

[162] Michael Burrows, Martin Abadi, and Roger Michael Needham. A logic of authentication. *Proceedings of the Royal Society of London. A. Mathematical and Physical Sciences*, 426(1871):233–271, 1989.

[163] Luca Viganò. Automated security protocol analysis with the avispa tool. *Electronic Notes in Theoretical Computer Science*, 155:61–86, 2006.

[164] Alessandro Armando, David Basin, Yohan Boichut, Yannick Chevalier, Luca Compagna, Jorge Cuéllar, P Hankes Drielsma, Pierre-Cyrille Héam, Olga Kouchnarenko, Jacopo Mantovani, et al. The avispa tool for the automated validation of internet security protocols and applications. In *International conference on computer aided verification*, pages 281–285. Springer, 2005.

[165] Avispa automated validation of internet security protocols and applications.

[166] H Hakan Kilinc and Tugrul Yanik. A survey of sip authentication and key agreement schemes. *IEEE Communications Surveys & Tutorials*, 16(2):1005–1023, 2014.

[167] Wendi B Heinzelman, Anantha P Chandrakasan, and Hari Balakrishnan. An application-specific protocol architecture for wireless microsensor networks. *IEEE Transactions on wireless communications*, 1(4):660–670, 2002.

[168] Sye Loong Keoh, Sandeep S Kumar, and Hannes Tschofenig. Securing the internet of things: A standardization perspective. *IEEE Internet of things Journal*, 1(3):265–275, 2014.

[169] Musab Ghadi, Lamri Laouamer, and Tarek Moulahi. Securing data exchange in wireless multimedia sensor networks: perspectives and challenges. *Multimedia Tools and Applications*, 75(6):3425–3451, 2016.

[170] Parvaneh Asghari, Amir Masoud Rahmani, and Hamid Haj Seyyed Javadi. Internet of things applications: A systematic review. *Computer Networks*, 148:241–261, 2019.

[171] Yasmine Harbi, Allaoua Refoufi, Zibouda Aliouat, and Saad Harous. Efficient end-to-end security scheme for privacy-preserving in iot. In *2019 International Conference on Networking and Advanced Systems (ICNAS)*, pages 1–6. IEEE, 2019.

[172] Serge Vaudenay. *A classical introduction to cryptography: Applications for communications security*. Springer Science & Business Media, 2006.

[173] Hamed Hellaoui, Mouloud Koudil, and Abdelmadjid Bouabdallah. Energy-efficient mechanisms in security of the internet of things: A survey. *Computer Networks*, 127:173–189, 2017.

[174] Kerry McKay, Lawrence Bassham, Meltem Sönmez Turan, and Nicky Mouha. Report on lightweight cryptography. Technical report, National Institute of Standards and Technology, 2016.

[175] Sohel Rana, Saddam Hossain, Hasan Imam Shoun, and Mohammod Abul Kashem. An effective lightweight cryptographic algorithm to secure resource-constrained devices. *International Journal of Advanced Computer Science and Applications*, 9(11), 2018.

[176] SN Sivanandam and SN Deepa. Genetic algorithms. In *Introduction to genetic algorithms*, pages 15–37. Springer, 2008.

[177] Akila Zirem and Mustapha Reda Senouci. Efficient lightweight chaotic secure communication system for wsns and iot. In *2018 International Conference on Smart Communications in Network Technologies (SaCoNeT)*, pages 43–48. IEEE, 2018.

[178] Hugo Krawczyk, Mihir Bellare, and Ran Canetti. Hmac: Keyed-hashing for message authentication, 1997.

[179] Danny Dolev and Andrew Yao. On the security of public key protocols. *IEEE Transactions on information theory*, 29(2):198–208, 1983.

[180] Shujun Li, Guanrong Chen, and Xuanqin Mou. On the dynamical degradation of digital piecewise linear chaotic maps. *International journal of Bifurcation and Chaos*, 15(10):3119–3151, 2005.

[181] Claude E Shannon. Communication theory of secrecy systems. *Bell system technical journal*, 28(4):656–715, 1949.

[182] George F Riley and Thomas R Henderson. The ns-3 network simulator. In *Modeling and tools for network simulation*, pages 15–34. Springer, 2010.

[183] Longteng Yi, Xiaojun Tong, Zhu Wang, Miao Zhang, Honghong Zhu, and Jing Liu. A novel block encryption algorithm based on chaotic s-box for wireless sensor network. *IEEE Access*, 7:53079–53090, 2019.

[184] Michael Armbrust, Armando Fox, Rean Griffith, Anthony D Joseph, Randy Katz, Andy Konwinski, Gunho Lee, David Patterson, Ariel Rabkin, Ion Stoica, et al. A view of cloud computing. *Communications of the ACM*, 53(4):50–58, 2010.

[185] Opeyemi Osanaiye, Shuo Chen, Zheng Yan, Rongxing Lu, Kim-Kwang Raymond Choo, and Mqhele Dlodlo. From cloud to fog computing: A review and a conceptual live vm migration framework. *IEEE Access*, 5:8284–8300, 2017.

[186] Xiong Li, Jianwei Niu, Saru Kumari, Fan Wu, Arun Kumar Sangaiah, and Kim-Kwang Raymond Choo. A three-factor anonymous authentication scheme for wireless sensor networks in internet of things environments. *Journal of Network and Computer Applications*, 103:194–204, 2018.

[187] Konstantinos Christidis and Michael Devetsikiotis. Blockchains and smart contracts for the internet of things. *IEEE Access*, 4:2292–2303, 2016.

[188] Loi Luu, Duc-Hiep Chu, Hrishi Olickel, Prateek Saxena, and Aquinas Hobor. Making smart contracts smarter. In *Proceedings of the 2016 ACM SIGSAC conference on computer and communications security*, pages 254–269, 2016.

[189] Yevgeniy Dodis, Leonid Reyzin, and Adam Smith. Fuzzy extractors: How to generate strong keys from biometrics and other noisy data. In *International conference on the theory and applications of cryptographic techniques*, pages 523–540. Springer, 2004.

[190] Thomas S Messerges, Ezzat A Dabbish, and Robert H Sloan. Examining smart-card security under the threat of power analysis attacks. *IEEE transactions on computers*, 51(5):541–552, 2002.

[191] David Basin, Sebastian Mödersheim, and Luca Vigano. Ofmc: A symbolic model checker for security protocols. *International Journal of Information Security*, 4(3):181–208, 2005.

[192] Ammar Awad Mutlag, Mohd Khanapi Abd Ghani, Net al Arunkumar, Mazin Abed Mohammed, and Othman Mohd. Enabling technologies for fog computing in healthcare iot systems. *Future Generation Computer Systems*, 90:62–78, 2019.

[193] Mohammad Aazam, Sherali Zeadally, and Khaled A Harras. Deploying fog computing in industrial internet of things and industry 4.0. *IEEE Transactions on Industrial Informatics*, 14(10):4674–4682, 2018.

[194] Mohamed Amine Ferrag, Lei Shu, Xing Yang, Abdelouahid Derhab, and Leandros Maglaras. Security and privacy for green iot-based agriculture: Review, blockchain solutions, and challenges. *IEEE Access*, 8:32031–32053, 2020.

[195] Debiao He, Neeraj Kumar, Jong-Hyouk Lee, and R Simon Sherratt. Enhanced three-factor security protocol for consumer usb mass storage devices. *IEEE Transactions on Consumer Electronics*, 60(1):30–37, 2014.