A REVIEW OF ECC BASED TWO-FACTOR AUTHENTICATION SCHEMES FOR WSN

by

Name: David

Student ID: 202118010418

Group: L5C-4

An assignment submitted in partial fulfillment
of the requirements for the module of

[Foundations of Security]

[Chengdu University of Technology]

Submitted To _____

DR. MD JAKIR HOSSAIN

DEPARTMENT OF COMPUTER SCIENCE

OXFORD BROOKES UNIVERSITY AND CHENGDU UNIVERSITY OF TECHNOLOGY COLLABORATIVE PROGRAMME

_____

Date 2024/5/11 _____

ABSTRACT

The rapid expansion of the Internet of Things (IoT) has led to an increasing reliance on wireless sensor networks (WSN) for a variety of applications, including environmental monitoring, intelligent transportation, and healthcare. Wireless sensor network technology, as an important part of the Internet of Things technology, makes it possible for the Internet of Things to perceive the world. However, the distributed nature and wireless communication mechanisms of wireless sensor networks make them vulnerable to many security threats such as data tampering, node capture, and denial of service attacks. Addressing these security concerns is critical to ensuring the privacy and security of users and the stability of the network. Therefore, in recent years, more and more literatures have proposed two-factor authentication schemes to ensure the security of WSN, claiming that their schemes can effectively balance security and computing performance, and can better adapt to the limited performance of devices in WSN.

This course work delves into the design and analysis of secure identity authentication schemes for WSNs, focusing on two-factor authentication schemes that have been proposed since 2022. These schemes aim to balance security with computational efficiency, a crucial aspect given the limited resources of sensor nodes in terms of compute, power, and memory. The study reviews three typical two-factor authentication schemes, providing a detailed mathematical analysis and comparing their security and performance parameters. The comparative analysis reveals that the scheme proposed by Li et al. in 2024 offers the most robust security and is well-suited for WSN devices with limited performance. It effectively enhances user anonymity and provides resistance against known attacks, including smart card theft and denial of service (DoS) attacks. The work also identifies areas for improvement, suggesting simplification of encryption operations, enhanced scalability, user-centric design, periodic security updates, and comprehensive performance evaluation to future-proof against emerging threats.

Key: ***ECC, WSN, two-factor authentication***

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGMENTS

The author wishes to express sincere appreciation to Teachers and their friends for their assistance in the preparation of this manuscript. In addition, special thanks to Rachel and James whose familiarity with the needs and ideas of the class was helpful during the early programming phase of this undertaking. Thanks also to the members of the school council for their valuable input.

Chapter 1

INTRODUCTION

Wireless sensor network (WSN), as an important part of the Internet of Things (IoT), has been widely studied and applied in recent years [1]. Wireless sensor network (WSN), as a key sensing technology, plays an increasingly important role in many fields such as environmental monitoring, intelligent transportation, and health care [2].

WSN is formed by a large number of sensor nodes connected to each other via Wi-Fi, Bluetooth, or ZigBee, which can collaborate to collect and transmit data to provide people with real-time information [3]. Wireless sensor networks include three types of participants: users, sensor nodes and gateways. The user initiates a session through the gateway. The sensor node has the functions of collecting, storing, calculating and uploading the original data. Gateways collect data by constantly communicating with individual sensor nodes [4].

However, the distributed nature and wireless communication mechanism of WSN also make it the target of various security threats, including data tampering, node capture, denial of service attacks, etc. First of all, because WSN transmits data wirelessly, it also faces various threats that ordinary wireless networks face, such as replay attacks, information leakage, denial of service attacks, and so on [5]. Secondly, in order to facilitate real-time information collection, sensor nodes in wireless sensor networks are mostly deployed in unattended public environments [6]. As a result, sensor node devices in WSN can easily be physically accessed or compromised by attackers. Moreover, due to the network connectivity of WSN, if one of the nodes is captured by an attacker, the attacker can not only use the node to obtain important information such as user information, but also use the captured device as a medium to attack other nodes in the network, thus damaging the entire system [7]. In addition, because WSN has limited resources in terms of compute, power, and memory. Based on the sensitivity and criticality of the data, the data must be protected by end-to-end services when it is transferred between entities outside WSN and WSN [8].

These problems not only threaten the stability and reliability of the network, but also pose a serious challenge to the privacy and security of users [3-5]. Therefore, it is very important to design a secure identity authentication scheme to ensure the security of data transmission between users and sensor nodes in wireless sensor networks.

Through the two-factor authentication scheme using authentication and key management techniques, many literatures claim that this can solve the security problems mentioned above and has better computational efficiency. Different entities in WSN can authenticate each other, thus effectively preventing various attacks.

In order to make readers quickly understand the research of two-factor authentication schemes in WSN in recent years, this course work mainly reviews and analyzes the two-factor authentication schemes in wireless sensor networks since 2022, and compares the security analysis and performance of similar schemes in recent years. The results show that in recent years, most two-factor authentication schemes can satisfy user anonymity and certain security, and strike a balance between security and computational efficiency.

The contributions of this course work are summarized as follows:

a) This course work reviews and details three typical two-factor certification schemes since 2022, and points out the best security scheme considered by this course work;
b) This course work analyzes and compares the performance parameters such as security and computational efficiency of similar schemes in recent years.
c) This course work has found that two-factor authentication schemes since 2022 generally have high security against commonly known attacks, while satisfying the limited performance of devices in WSN.

This course work is organized as follows: Chapter 2 is related work and reviews the research on authentication schemes in WSN. Chapter 3 is a preliminary and contains some of the concepts necessary to understand Chapter 4. Chapter 4 reviews and mathematically analyzes three typical two-factor authentication schemes. In Chapter 5, the scheme introduced in this course work is compared with the similar schemes, and the results of security, communication cost and computational cost are given. Chapter 6 summarizes the full text.

Chapter 2

RELATED WORKS

Since the first identity authentication scheme was proposed in 1981[9], many authentication schemes claiming that they balance WSN efficiency and security have been proposed and studied in the literature, but many schemes are considered to have security risks.

In 2013, in order to guarantee the security of data transmission between sensor nodes and users in WSN, Benenson et al. [10] first proposed a solution to this problem. Although Benenson et al. 's scheme provides some security, it is inefficient and has low practical application value. Later, Jiang et al. [11] proposed an enhancement scheme using self-certified public key cryptography (PKC). However, this scheme has a huge storage cost problem of public key pairs in memory.

In recent years, lightweight authentication schemes have been studied by many researchers in order to solve the problem of limited resources of sensor devices in WSN [12][13]. The lightweight authentication scheme mainly includes Elliptic curve cryptography (ECC), chaotic map and one-way hash function construction [3]. In 2019, Sharif et al. [12] proposed a lightweight and efficient identity authentication scheme based on WSN in the Internet of things environment, and claimed that their scheme can resist known attacks. In 2020, Chen et al. [13] pointed out that there were security vulnerabilities in the scheme proposed by Sharif et al. and proposed an improved scheme. Not only the legitimate users cannot access the system due to the system design problems, but also the potential password leakage problem caused by the immutability of passwords.

However, lightweight authentication schemes generally have the problem of insufficient security because of their low computational load. In 2009, Das et al. [14] proposed the WSN two-factor identity authentication scheme for the first time in order to satisfy both low computing load and security, which has the characteristics of high efficiency and resistance to various attacks. A year later, Khan et al. [15] pointed out that Das's scheme was vulnerable to impersonation attacks and insider attacks and proposed an improved scheme to solve the problem. Subsequently, Vaidya et al. [16]

pointed out that Khan et al. 's scheme still did not effectively solve the security risks in 2016.In 2022, Hu et al. [3] claimed that Chen et al. 's improved scheme [13] still had security risks and proposed an enhanced two-factor identity authentication scheme to overcome the security weaknesses of Chen et al.' s scheme. They claim that the security enhancement scheme is universal, secure and computationally efficient, and it is the first WSN authentication scheme that maintains user anonymity and two-factor security. However, in 2024, Li et al. [5] pointed out that Hu et al. 's two-factor identity authentication scheme was difficult to resist DOS and smart card theft attacks, and proposed an improved scheme to solve the security risks of Hu et al.' s scheme and claimed that its improved scheme met the limited power characteristics of WSN devices.

In this course work, we focus on analyzing three typical two-factor certification schemes from 2022 and comparing their performance with similar schemes. The performance comparison results show that the latest two-factor authentication scheme [5] has the best robustness and security among similar authentication schemes, and is suitable for wireless sensor network devices with limited performance.

Chapter 3

PRELIMINARIES

In order to facilitate the understanding of the scheme introduced in this course work. This section provides symbolic definitions used in the security scenarios described in this course work, along with the associated mathematical foundations.

3.1 ECC (Elliptic Curve cryptography)

Elliptic Curve Cryptography (ECC) is a specialized form of public-key cryptography that leverages the Elliptic Curve Discrete Logarithm Problem (ECDLP). Unlike traditional methods like RSA, ECC provides equivalent security with a much shorter key length, making it particularly suitable for environments with limited computational resources. The concept of ECC was prognosticated by Victor Miller in 1985, with Neal Koblitz also contributing to its development in the same year.

To define an elliptic curve for use in cryptography over a finite field Fp, six parameters are typically used: T= (p, a, b, G, n, h). Here, p is the prime number defining the field, and a and b are the coefficients of the elliptic curve equation. G is the base point on the curve, n is the order of the base point G, and h is the cofactor representing the ratio of the number of points on the curve to the order of the base point, divided by n. The selection of these parameters is critical for the security of the encryption.

The ECC encryption and decryption process can be summarized as follows:

- User A chooses an elliptic curve T= (p, a, b, G, n, h).
- User A chooses a private key k and computes the corresponding public key K as K=kG.
- User A shares the curve parameters T and the public key K, along with the base point G, with User B.
- Upon receiving this information, User B encodes the plaintext into a point M on the curve T and chooses a random integer r such that r is less than n.
- User B then calculates two points on the curve: C1 as C1=M+rK and C2 as C2=rG.

- User B forwards the points C1 and C2 to User A.
- User A, upon receiving C1 and C2, computes the original message point M by calculating C1-kC2, which simplifies to M because C1-kC2=M+rK-k(rG)=M+rK-r(kG)=M.
- User A then decodes the point M to retrieve the plaintext.

In this encrypted communication, if an eavesdropper, Tom H, intercepts the conversation, he can only observe the curve parameters T and the points K, G, C1, C2. It is computationally difficult for Tom H to determine the private key k from K and G or the random integer r from C2 and G. Hence, Tom H cannot decipher the plaintext exchanged between User A and User B.

3.2 Hash Functions

Hash functions are fundamental components in cryptography, known for their ability to ensure data integrity and verify authenticity. A hash functions

$$H: \{0,1\}^{*} \rightarrow \{0,1\}^{n}$$

is a mathematical function that maps an arbitrary length input (message) to a fixed length output (hash value), where n is the bit length of the hash value. The standard hash function should have the characteristics of unipolarity, collision resistance, uniformity and rapidity.

Unipolarity: It should be computationally infeasible to find an input mm given a hash value h, i.e., $h=H(m)$, but one cannot deduce mm from h alone.
Collision resistance: It should be computationally infeasible to find two distinct inputs $m1$ and $m2$ such that $H(m1) = H(m2)$, known as a collision.
Uniformity: The hash function should uniformly map inputs to the output space, meaning that each output bit should be independently and uniformly distributed.
Efficiency: Hash functions should be computationally efficient to allow for processing large volumes of data.
Mathematical Expression: A simple representation of a hash function can be expressed as:

$$H(m) = \left(\sum_{i=1}^{l} m_i \times 2^{ni} \right) \bmod p$$

where m is the input message, l is the length of the message, m_i is the i-th bit of the message, n_i is the corresponding bit shift, and p is a large prime number.

The design and analysis of hash functions is a complex field in cryptography. In practical applications, choosing the right hash function is very important to ensure the security of the system.

3.3 XOR Operation

XOR operation is a simple binary operation, which is often used in cryptography for data encryption and error detection. Its operation rules are as follows: For two bits, the result is 0 when the two bits are the same, and 1 when they are different.

The XOR operation is used in symmetric encryption algorithms, particularly in stream ciphers, where a pseudo-random keystream is XORed with the plaintext to produce the ciphertext. It is also used in the one-time pad, a theoretically unbreakable encryption scheme where the plaintext is XORed with a truly random key of the same length. For bit strings, the XOR operation can be expressed as a modulo 2 additions (since XOR is equivalent to addition without carrying in binary):

$$A \oplus B = (A + B) \bmod 2.$$

3.4 Notations Definition

Notations	Description
GWN	Gateway node, is a trusted node in the system. Users and sensor nodes use GWN to authenticate each other and negotiate session keys
U	User, an individual that performs authentication and data exchange in the system.
S	Sensor node, a sensor node in a wireless sensor network that collects and transmits data
ID, ID_{GWN}, SID	Identifiers that uniquely identify users, GWN and sensor node
PID	Pseudonymous identifier for users

PW	Indicates the password of the user
K_{GU}, K_{GS}	Private keys, known only to GWN
$\boldsymbol{K_{GWN-U}}$, $\boldsymbol{K_{GWN-S}}$	Shared secret-key among Gateway and User, Shared secret-key among gateway and sensor
K	Temporary session key for secure communication between user and sensor node.
$\boldsymbol{PTC_i}$	Potential temporal credentials of User
TC, TC_i, TC_j	Temporal Credential used in the authentication of users or sensor nodes.
$\boldsymbol{TS_i}$	Time-stamp value
$\boldsymbol{TE_i}$	Indicates the Expiration Time of the user's temporary certificate.
$\boldsymbol{DID_i}$, $\boldsymbol{DID_{GWN}}$	Dynamic identity of User and sensor
P	Base point of an elliptic curve
h ()	Safe one-way hash function
$\oplus$	An XOR operation used during data encryption or authentication
$\|\|$	Concatenation
$\boldsymbol{\Delta T}$	Message Transmission delay

TABLE.1 Notations Definition

Chapter 4

LITERATURE ANALYSIS

This section introduces the two-factor authentication schemes of Hu et al., li et al., and Chander et al., by means of mathematics. Finally, it is pointed out that li et al. 's scheme is the best one at present, and the explanation is given.

4.1 Review of Hu et al.'s Scheme

In Hu et al.'s scheme, users and sensor nodes complete mutual authentication and agree on the session key with the help of GWN. The scheme consists of four phases: the initialization phase, the registration phase, the login and key protocol phase, and the password and expiration time update phase. The symbols involved in this scheme and their definitions are shown in chapter 3.

4.1.1 Initialization phase

GWN (Gateway node) chooses two random numbers K_{GU} and K_{GS} as its private key, and chooses a generator P on an elliptic curve and a hash security function h ().
Finally, the public key of GWN, P_{pub}, is generated, and the formula is as follows:

$$P_{pub} = K_{GU} \cdot P$$

4.1.2 Registration Phase

The current phase includes the user registration phase and the sensor node registration phase. During this phase, data is transmitted through a secure channel.

4.1.2.1 User Registration

New users must first register on the gateway before they can access WSN and use related services. The detailed steps of the user registration phase are as follows:

Step 1 U_i chooses ID_i and PW_i generates a random number r_i, and computes

$$A_i = h(\mathrm{ID}_i \parallel \mathrm{PW}_i \parallel r_i).$$

Then, U_i sends a message $\{\mathrm{ID}_t, A_t\}$ to the GWN by a secure channel.

Step 2 Once GWN receives a message from U_i, it chooses an expiration time TE_i for the temporary credentials of U_i.GWN computes the public key

$$P_{\mathrm{pub}} = K_{\mathrm{GU}} \cdot P$$

and U_i's temporal credential

$$\mathrm{TC}_i = h(\mathrm{ID}_i \parallel \mathrm{ID}_{\mathrm{GWN}} \parallel K_{\mathrm{GU}} \parallel \mathrm{TE}_i).$$

GWN will store $\{\mathrm{ID}_{\mathrm{Gws}}, \mathrm{TE}_i, P_{\mathrm{pab}}, h(\cdot), \mathrm{PTC}_i\}$ in a smart card SC and send SC to U_i by a secure channel

Step 3 U_i computes $\mathrm{TC}_i = \mathrm{PTC}_i \oplus A_i, B_i = \mathrm{TC}_i \oplus h(\mathrm{ID}_i \parallel \mathrm{PW}_i)$,and stores $\{B_i\}$ in SC.

4.1.2.2 Registering a Sensor Node

Registering a sensor node involves the node registering with GWN, which occurs only once. This registration phase includes the following steps:

Step 1 GWN chooses an identity SID, for the sensor node S_j and computes

$$\mathrm{TC}_j = h(K_{\mathrm{Gs}} \parallel \mathrm{SID}_j).$$

GWN then sends a message $\{\mathrm{TC}_j,\mathrm{SID}_j\}$ to S_j by a secure channel.

Step 2 S_j receives the message from GWN and stores$\{\mathrm{TC}_j, \mathrm{SID}_j\}$.

4.1.3 Login and Key Agreement Phase

During this stage, the U_i and S_j, assisted by GWN, agree on the session key, executing a mutually authenticated key agreement to ensure further secure communication between them, as illustrated in Figure 1.

The specific steps are as follows:

Step 1 U_i inserts his/her smart card (SC) and enters his/her identity ID_i and password PW_i. SC then generates two random numbers N_1 and x_1 based on the stored information and the extracted

$$\mathrm{TC}_i = B_i \oplus h(\mathrm{ID}_i \parallel \mathrm{PW}_i)$$

entered by U_i, and the system computes

$$T_1 = x_1 \cdot P,$$

$$T_2 = (\mathrm{ID}_i \parallel \mathrm{TE}_i \parallel \mathrm{SID}_j \parallel N_1) \oplus h(x_1 \cdot P_{\mathrm{pub}}),$$

where $P_{\mathrm{pub}} = K_{\mathrm{GU}} \cdot P$ is the public key of GWN,

$$T_3 = h(T_1 \parallel \mathrm{ID}_i \parallel \mathrm{ID}_{\mathrm{GWN}} \parallel \mathrm{TC}_i \parallel N_1 \parallel \mathrm{TE}_i \parallel \mathrm{SID}_j).$$

U_i sends a login request message $M_1 = \{T_1, T_2, T_3\}$ to GWN.

Step 2 Upon receiving the message $M_1 = \{T_1, T_2, T_3\}$ from U_i ,GWN computes

$$\mathrm{ID}_i \parallel \mathrm{TE}_i \parallel \mathrm{SID}_j \parallel N_1 = T_2 \oplus h(K_{\mathrm{GU}} \cdot T_1)$$

and verifies TE_i. If invalid, GWN will reject the login request of U_i. If valid, GWN computes

$$\mathrm{TC}_i = h(\mathrm{ID}_i \parallel \mathrm{ID}_{\mathrm{GWN}} \parallel K_{\mathrm{GU}} \parallel \mathrm{TE}_i),$$

$$T_3^* = h(T_1 \parallel \mathrm{ID}_i \parallel \mathrm{ID}_{\mathrm{GWN}} \parallel \mathrm{TC}_i \parallel N_1 \parallel \mathrm{TE}_i \parallel \mathrm{SID}_j),$$

and checks $T_3^* = T_3$. If incorrect, GWN terminates the current phase. If correct, GWN generates three random numbers N_2, x as well as x_2 and computes

$$\mathrm{TC}_j = h(K_{\mathrm{GS}} \parallel \mathrm{SID}_j),$$

$$T_4 = x_2 \oplus h(\mathrm{TC}_j \parallel N_2 \parallel \mathrm{ID}_{\mathrm{GWN}}),$$

$$T_5 = h(\mathrm{ID}_i \parallel \mathrm{TE}_i \parallel x) \oplus h(N_2 \parallel \mathrm{TC}_j),$$

$$T_6 = h(T_1 \parallel h(\mathrm{ID}_i \parallel \mathrm{TE}_i \parallel x) \parallel x_2 \parallel N_2).$$

Then, GWN sends a message $M_2 = \{T_1, T_4, T_5, T_6, N_2\}$ to S_j by a secure channel.

Step 3 Upon receiving the message from GWN,S_j recovers

$$x_2 = T_4 \oplus h\big(\mathrm{TC}_j \parallel N_2 \parallel \mathrm{ID}_{\mathrm{GWN}}\big),$$

$$h(\mathrm{ID}_i \parallel \mathrm{TE}_i \parallel x) = T_5 \oplus h\big(N_2 \parallel \mathrm{TC}_j\big),$$

and computes

$$T_6^* = h(T_1 \parallel h(\mathrm{ID}_i \parallel \mathrm{TE}_i \parallel x) \parallel x_2 \parallel N_2).$$

Then, S_j verifies $T_6 *= T_6$. If correct,S_j generates two random numbers N_3, x_3, and computes

$$\mathrm{SK} = h\big(h(\mathrm{ID}_i \parallel \mathrm{TE}_i \parallel x) \parallel \mathrm{SID}_j \parallel x_3 \cdot T_1 \parallel T_1 \parallel T_7\big),$$

$$T_8 = h(\mathrm{SK} \parallel N_3),$$

$$T_9 = (T_8 \parallel T_7 \parallel N_3) \oplus h\big(\mathrm{TC}_j \parallel N_2\big),$$

$$T_{10} = h\big(\mathrm{TC}_j \parallel T_7 \parallel N_2 \parallel T_8\big),$$

and transmits a message $M_3 = \{T_9, T_{10}\}$ to GWN.

Step 4 After GWN receiving the message from S_j,it extracts

$$T_8 \parallel T_7 \parallel N_3 = T_9 \oplus h(\mathrm{TC} \parallel N_2)$$

and computes

$$T_{10} *= h\big(\mathrm{TC}_j \parallel T_7 \parallel N_2 \parallel T_8\big).$$

Then, GWN verifies $T_{10}^* = T_{10}$. If incorrect, GWN will terminate the current phase immediately. If correct, GWN computes

$$T_{11} = (T_8 \parallel N_1 \parallel T_7 \parallel N_3 \parallel x) \oplus h(N_1 \parallel \mathrm{TC}_i),$$

and sends the message $M_4 = \{T_{11}\}$ to U_i.

Fig. 1. Login and key agreement phase of the *Hu et al.'s* Scheme.

Step 5 Upon receiving M_4, U_i extracts

$$T_8 \parallel N_1 \parallel T_7 \parallel N_3 \parallel x = T_{11} \oplus h(N_1 \parallel \mathrm{TC}_i)$$

and computes the session key

$$\mathrm{SK} = h\big(h(\mathrm{ID}_i \parallel \mathrm{TE}_i \parallel x) \parallel \mathrm{SID}_j \parallel x_3 \cdot T_1 \parallel T_1 \parallel T_7\big),$$

$$T_8^{\,*} = h\big(\mathrm{SK} \parallel N_3\big).$$

U_i verifies if $T_8^* = T_8$. If it does, indicating successful session key establishment between U_i and S_j.

4.1.4 Password and Expiration Time Update Phase

If U_i wishes to update or change his/her password, he/she inserts his/her smart card SC and enters ID_i, PW_i. After that, SC will calculate

$$B_i^{\mathrm{new}} = B_i \oplus h(\mathrm{ID}_i \parallel \mathrm{PW}_i) \oplus h\big(\mathrm{ID}_i \parallel \mathrm{PW}_i^{\mathrm{new}}\big),$$

and then replaces B_i with B_i^{new}.

If GWN needs to update the expiration time TE_i, for TC_i,GWN can reselect a new TE_i' and recalculate

$$\mathrm{TC}_i' = h(\mathrm{ID}_i \parallel \mathrm{ID}_{\mathrm{GWN}} \parallel K_{\mathrm{GU}} \parallel \mathrm{TE}_i'),$$

$$T_{11}' = (T_8 \parallel N_1 \parallel T_7 \parallel N_3 \parallel \mathrm{TC}_i' \parallel \mathrm{TE}_i') \oplus h(N_1 \parallel \mathrm{TC}_i)$$

in Step 4 of the login and key agreement phase. Then, U_i can extract TE_i' and TC_i' from T_{11}, update B_i and TE_i in his/her own smart card.

4.2 Review of Li et al.'s Scheme

In 2024, Li et al. pointed out that Hu et al. 's scheme had security risks. For example, Hu et al. 's scheme does not protect against smart card theft attacks, DOS attacks, and the inability to implement effective mutual authentication and key protocols at the registration stage. Therefore, Li

et al. proposed an improved scheme based on the security scheme proposed by Hu et al. to solve the above problems. Firstly, the scheme adds the dynamic user pseudo-identity in the user registration stage, and transmits the identity information through the public channel. Secondly, the scheme includes key authentication in the user login stage, that is, the smart card verifies whether the current user is a legitimate registered user. Finally, the scheme adds some necessary identity information to the transmitted message. This ensures that when GWN receives a message from a sensor node, it is clear which sensor node is communicating with which user.

The scheme meets the requirement of mutual authentication and effectively enhances the anonymity of users. In addition, the scheme can not only resist the above-mentioned stolen smart card attacks and DOS attacks, but also solve the problem that GWN cannot extract key values. For brevity, this section covers only the initialization phase, the registration phase, and the login and key protocol phases. The specific steps are as follows:

4.2.1 Initialization Phase

GWN chooses an additive group G of order q and a generator P of G on an elliptic curve E. GWN chooses two private keys $K_{\mathrm{GU}} \in Z_q^*$ and $K_{\mathrm{GS}} \in Z_q^*$, and computes its public key

$$P_{\mathrm{pub}} = K_{\mathrm{GU}} \cdot P.$$

4.2.2 Registration Phase

Any user or sensor node wishing to communicate with GWN must undergo registration to ensure subsequent communication security. This phase consists of user registration and sensor node registration.

4.2.2.1 Registration of users

Before accessing an IoT service and communicating with sensor nodes, new users must register with GWN and acquire their smart card (SC) through a secure channel. GWN stores the registration information for identity verification during the login phase. In Fig.2, this phase is divided into three steps and the process is as follows:

Step 1 User U_i chooses his/her unique identity ID_i and password PW_i, generates a random number $r_i \in Z_q^*$, and computes A_i and pseudo-identity PID_i using equation (1). Then, U_i sends a registration message $\{\mathrm{ID}_i, A_i, \mathrm{PID}_i\}$ to the GWN by a secure channel.

$$\begin{cases} A_i = h(\mathrm{ID}_i \parallel \mathrm{PW}_i \parallel r_i) \\ \mathrm{PID}_i = h(\mathrm{ID}_i \parallel r_i) \end{cases} \tag{1}$$

Step 2 GWN receives the registration message from U_i and chooses an expiration time TE_i for U_i. Then, GWN computes its own public key P_{pub}, the user's temporary credentials TC_i and PTC_i using equation (2).GWN stores $\{\mathrm{ID}_i, \mathrm{TE}_i, \mathrm{PID}_i\}$ into its own database and embeds $\{\mathrm{ID}_{\mathrm{GWN}}, \mathrm{TE}_i, P_{\mathrm{pub}}, h(\cdot), \mathrm{PTC}_i\}$ into a smart card SC, which is then issued to U_i via a secure channel

$$\begin{cases} P_{\mathrm{pub}} = K_{\mathrm{GU}} \cdot P \\ \mathrm{TC}_i = h(\mathrm{ID}_i \parallel \mathrm{PID}_i \parallel \mathrm{ID}_{\mathrm{GWN}} \parallel K_{\mathrm{GU}} \parallel \mathrm{TE}_i) \\ \mathrm{PTC}_i = \mathrm{TC}_i \oplus A_i \end{cases} \tag{2}$$

Step 3 Upon receiving the SC, U_i computes the values required for authentication in the next stage using equation (3). Subsequently, U_i stores $\{A_i, B_i, C_i\}$ in SC and removes PTC_i. At this point, the values stored in SC are $\{\mathrm{ID}_{\mathrm{GWN}}, \mathrm{TE}_i, P_{\mathrm{pub}}, h(\cdot), A_i, B_i, C_i\}$.

$$\begin{cases} \mathrm{TC}_i = \mathrm{PTC}_i \oplus A_i \\ B_i = \mathrm{TC}_i \oplus h(\mathrm{ID}_i \parallel \mathrm{PW}_i \parallel \mathrm{TE}_i) \\ C_i = r_i \oplus h(\mathrm{ID}_i \parallel \mathrm{PW}_i) \end{cases} \tag{3}$$

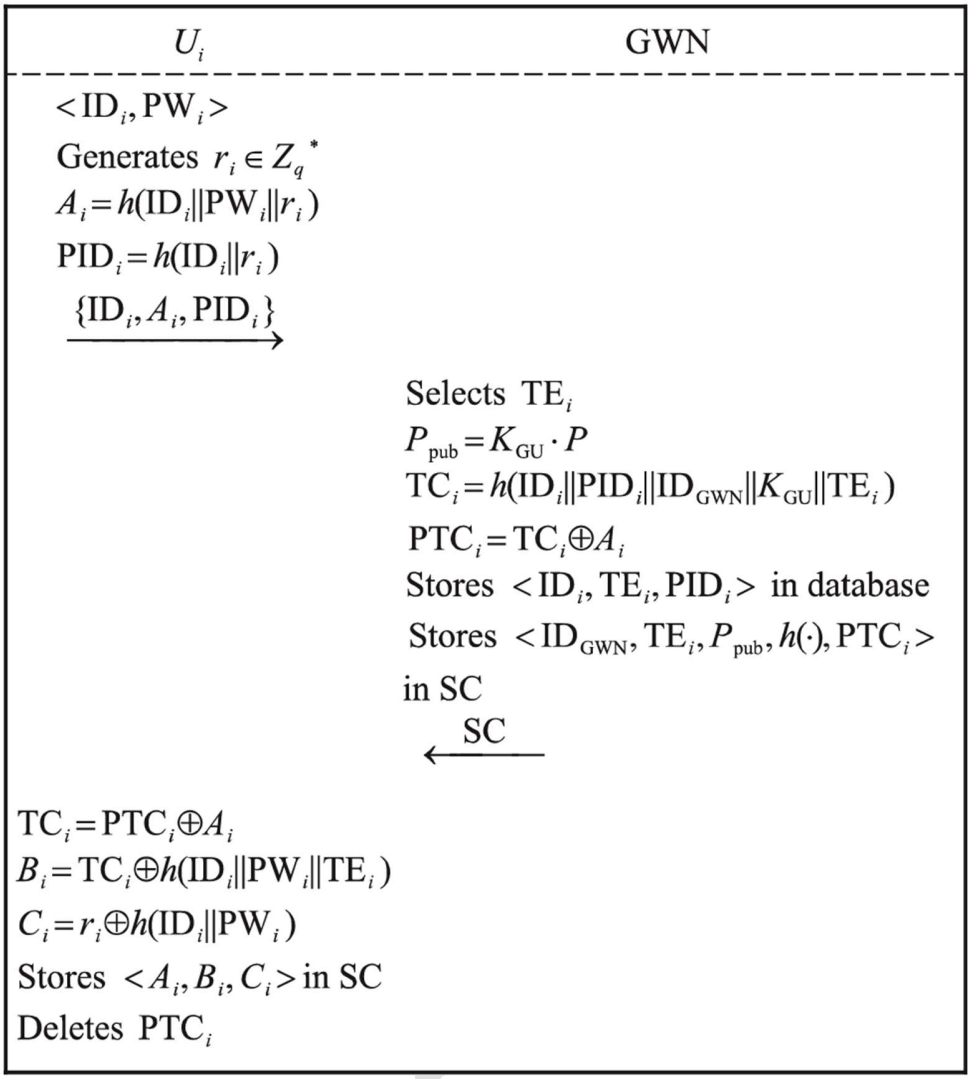

Fig. 2. Registration of users of the *Li et al.'s* Scheme.

4.2.2.2 Registration of sensor nodes

Sensor nodes interacting with GWN for the first time must undergo registration. The registration process for sensor nodes, as depicted in Fig. 3, involves the following steps:

Step 1 Upon installation, for a newly deployed sensor node S_j, GWN selects an identity SID_j for it and computes the temporary credential TC_j as:

$$\mathrm{TC}_j = h\left(K_{\mathrm{GS}} \| \mathrm{SID}_j\right).$$

Step 2 S_j receives a message $\{\mathrm{SID}_j, \mathrm{TC}_j\}$ sent from GWN via a secure channel and stores $\{\mathrm{SID}_i, \mathrm{TC}_i\}$ in memory.

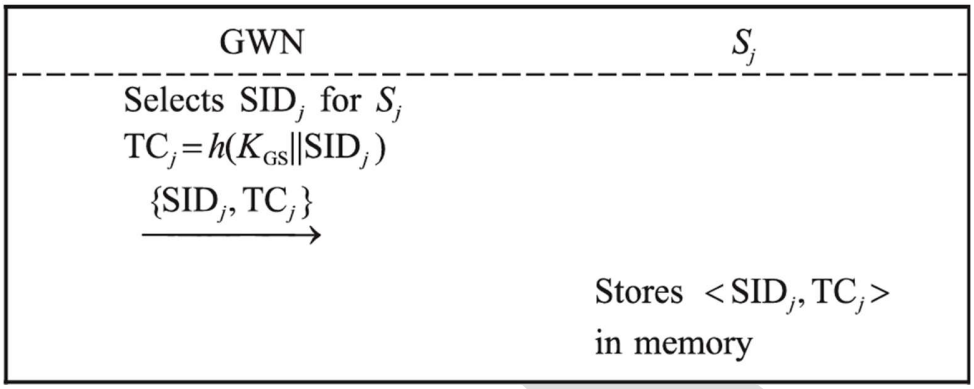

Fig. 3. Registration of *sensor nodes* of the *Li et al.'s* Scheme.

4.2.3 Login and Key Agreement Phase

In this stage, the legally registered user U_i can share a session key with registered sensor node S_j that needs to register via GWN. As depicted in Fig.4, U_i and S_j authenticate each other and create a session key for secure communication with GWN's assistance. The detailed steps are outlined below.

Step 1 U_i inserts the smart card SC and inputs his/ her identity ID_i and password PW_i. Then, SC utilizes equation (4) for the calculation.

$$\begin{cases} r_i^* = C_i \oplus h(\mathrm{ID}_i \parallel \mathrm{PW}_i) \\ A_i^* = h(\mathrm{ID}_i \parallel \mathrm{PW}_i \parallel r_i^*) \end{cases} \tag{4}$$

SC verifies $A_i^* = A_i$. If unsuccessful, SC will reject the user's login request. If successful, it indicates that this user is the legitimate holder of SC, and also indicates $r_i^* = r_i$. After that, SC recovers the temporary credentials TC_i for U_i and computes pseudo-identity PID_i using equation (5).

$$\begin{cases} \mathrm{TC}_i = B_i \oplus h(\mathrm{ID}_i \parallel \mathrm{PW}_i \parallel \mathrm{TE}_i) \\ \mathrm{PID}_i = h(\mathrm{ID}_i \parallel r_i) \end{cases} \tag{5}$$

Subsequently, SC chooses two random numbers $N_1 \in Z_q^*, x_1 \in Z_q^*$ and computes the values F_1, F_2, F_3 using equation (6). U_i sends message $M_1 = \{F_1, F_2, F_3, \mathrm{PID}_i\}$ to GWN via a public channel.

$$\begin{cases} F_1 = x_1 \cdot P \\ F_2 = (\mathrm{ID}_i \parallel \mathrm{TE}_i \parallel \mathrm{SID}_j \parallel N_1) \oplus h(x_1 \cdot P_{\mathrm{pub}}) \\ F_3 = h(F_1 \parallel \mathrm{ID}_i \parallel \mathrm{ID}_{\mathrm{GWN}} \parallel \mathrm{TC}_i \parallel N_1 \parallel \mathrm{TE}_i \parallel \mathrm{SID}_j) \end{cases} \tag{6}$$

Step 2 Upon receiving message M_1 from U_i, GWN extracts the values needed for subsequent authentication according to equation (7).

$$\mathrm{ID}_i \parallel \mathrm{TE}_i \parallel \mathrm{SID}_j \parallel N_1 = F_2 \oplus h(K_{\mathrm{GU}} \cdot F_1) \tag{7}$$

GWN verifies TE_i's effectiveness. If it failed, GWN rejects U_i's login request. If successful, GWN computes the TC_i and $F_3 *$ using equation (8).

$$\begin{cases} \mathrm{TC}_i = h(\mathrm{ID}_i \parallel \mathrm{PID}_i \parallel \mathrm{ID}_{\mathrm{GWN}} \parallel K_{\mathrm{GU}} \parallel \mathrm{TE}_i) \\ F_3 *= h(F_1 \parallel \mathrm{ID}_i \parallel \mathrm{ID}_{\mathrm{GWN}} \parallel \mathrm{TC}_i \parallel N_1 \parallel \mathrm{TE}_i \parallel \mathrm{SID}_j) \end{cases} \tag{8}$$

GWN verifies if $F_3 *= F_3$. If unsuccessful, GWN rejects U_i's login reques. Otherwise, it indicates that U_i is a legitimate user who has registered with GWN. After that, GWN chooses three random numbers $N_2 \in Z_q^*, x \in Z_q^*, x_2 \in Z_q^*$ and computes the temporary credentials TC_j of S_j and three values F_4, F_5, F_6 according to equation (9).

$$\begin{cases} \mathrm{TC}_j = h(K_{\mathrm{GS}} \parallel \mathrm{SID}_j) \\ F_4 = x_2 \oplus h(\mathrm{TC}_j \parallel N_2 \parallel \mathrm{ID}_{\mathrm{GWN}}) \\ F_5 = h(\mathrm{ID}_i \parallel \mathrm{TE}_i \parallel x) \oplus h(N_2 \parallel \mathrm{TC}_j) \\ F_6 = h(F_1 \parallel h(\mathrm{ID}_i \parallel \mathrm{TE}_i \parallel x) \parallel x_2 \parallel N_2) \end{cases} \tag{9}$$

Finally, GWN sends a message $M_2 = \{F_1, F_4, F_5, F_6, N_2, \mathrm{PID}_i\}$ to S_j via a public channel.

Step 3 Upon receiving the message M_2, S_j computes a series of values according to equation (10).

$$\begin{cases} x_2 = F_4 \oplus h(\mathrm{TC}_j \parallel N_2 \parallel \mathrm{ID}_{\mathrm{GWN}}) \\ h(\mathrm{ID}_i \parallel \mathrm{TE}_i \parallel x) = F_5 \oplus h(N_2 \parallel \mathrm{TC}_j) \\ F_6 *= h(F_1 \parallel h(\mathrm{ID}_i \parallel \mathrm{TE}_i \parallel x) \parallel x_2 \parallel N_2 \end{cases}$$

(10)

S_j verifies if $F_6 *= F_6$ holds or not. If not, the current phase is aborted. Otherwise, S_j generates two random numbers $N_3 \in Z_q^*$, $x_3 \in Z_q^*$ and computes the session key SK_{ji} and a set of values using equation (11).

$$\begin{aligned} F_7 &= x_3 \cdot P \\ \mathrm{SK}_{ji} &= h(\mathrm{PID}_i \parallel \mathrm{SID}_j \parallel x_3 \cdot F_1 \parallel F_1 \parallel F_7) \\ F_8 &= h(\mathrm{SK}_{ji} \parallel N_3) \\ F_9 &= (F_8 \parallel F_7 \parallel N_3) \oplus h(\mathrm{TC}_j \parallel \mathrm{PID}_i) \\ F_{10} &= h(\mathrm{TC}_j \parallel F_7 \parallel N_3 \parallel F_8) \end{aligned}$$

(11)

S_j sends a message $M_3 = \{F_9, F_{10}, \mathrm{SID}_j, \mathrm{PID}_i\}$ to GWN via a public channel.

Step 4 Upon receiving M_3, GWN extracts the values to be used for the subsequent operation and the validation value $F_{10} *$ using equation (12).

$$\begin{cases} F_8 \parallel F_7 \parallel N_3 = F_9 \oplus h(\mathrm{TC}_j \parallel \mathrm{PID}_i) \\ F_{10} *= h(\mathrm{TC}_j \parallel F_7 \parallel N_3 \parallel F_8) \end{cases}$$

(12)

GWN verifies if $F_{10}^* = F_{10}$ holds or not. If not, this session is aborted immediately. Otherwise, GWN updates PID_i using equation (13) and computes some values.

$$\begin{cases} \mathrm{PID}_i^{\mathrm{new}} = \mathrm{PID}_i \oplus h(N_3 \parallel \mathrm{TC}_i) \\ F_{11} = \mathrm{PID}_i^{\mathrm{new}} \oplus h(\mathrm{ID}_{\mathrm{GWN}} \parallel \mathrm{TC}_i \parallel N_1) \\ F_{12} = h(\mathrm{PID}_i^{\mathrm{new}} \parallel \mathrm{TC}_i) \\ F_{13} = (F_8 \parallel F_7 \parallel N_3) \oplus h(\mathrm{PID}_i^{\mathrm{new}}) \end{cases}$$

(13)

GWN then sends a message $M_4 = \{F_{11}, F_{12}, F_{13}\}$ to U_i via a public channel.

Step 5 Upon receiving the message from GWN, U_i extracts the updated pseudo-identity $\mathrm{PID}_i^{\mathrm{new}*}$ and computes $F_{12} *$ using equation (14).

$$\begin{cases} \mathrm{PID}_i{}^{\mathrm{ncw}*} = F_{11} \oplus h(\mathrm{ID}_{\mathrm{GWN}} \parallel \mathrm{TC}_i \parallel N_1) \\ F_{12} *= h(\mathrm{PID}_i{}^{\mathrm{ncw}*} \parallel \mathrm{TC}_i) \end{cases}$$

(14)

U_i verifies if $F_{12} *= F_{12}$ holds or not. If not, the session is terminated. Otherwise, it indicates successful authentication and message accuracy, and also means that $\mathrm{PID}_i^{\mathrm{new}*} = \mathrm{PID}_i^{\mathrm{new}}$. Then, U_i recovers and computes some values using equation (15).

$$\begin{cases} F_8 \parallel F_7 \parallel N_3 = F_{13} \oplus h(\mathrm{PID}_i^{\mathrm{new}}) \\ \mathrm{SK}_{ij} = h\left(\mathrm{PID}_i \parallel \mathrm{SID}_j \parallel x_1 \cdot F_7 \parallel F_1 \parallel F_7\right) \\ F_8 *= h\left(\mathrm{SK}_{ij} \parallel N_3\right) \end{cases}$$

(15)

U_i verifies if $F_8^* = F_8$ holds or not. If not, the session is terminated. Otherwise, it indicates successful negotiation of a session key between U_i and S_j for securing subsequent communications.

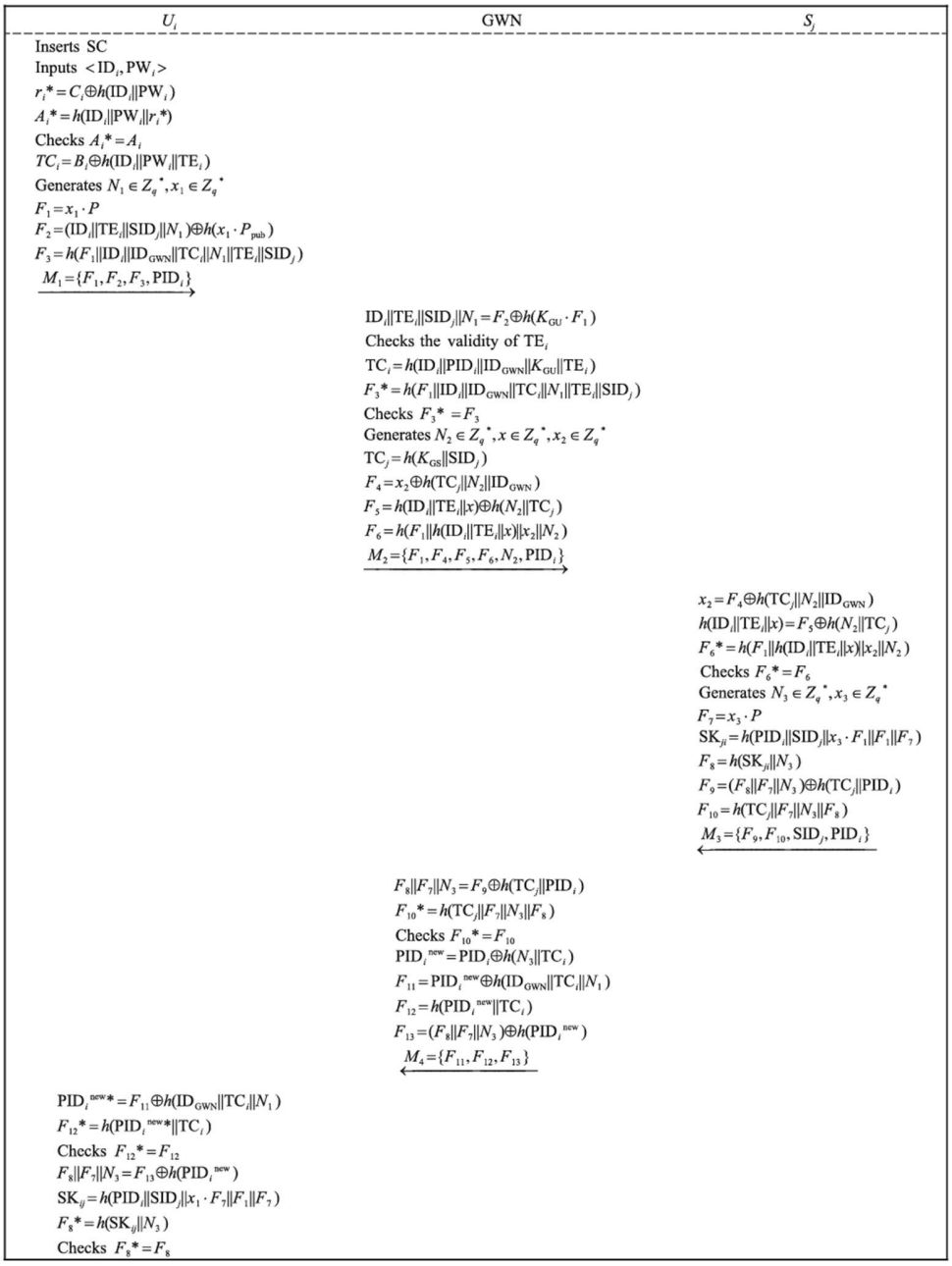

Fig. 4. Login and key agreement phase of the *Li et al.'s* Scheme.

4.3 Review of Chander et al.'s Scheme

This course work proposes a 2-factor verification scheme based on ECC and biometrics for resource-constrained WSN, which realizes unique security functions and protocol design logic.

4.3.1 Registration Phase

The following steps outline the registration process between a user U_i and the GWN for accessing remote sensor information from the WSN.

Step 1: U_i initiates registration by choosing an identity ID_i, password PW_i, and generating a random number b_i. A biometric feature Gen () =<> engages the Gen function using the fuzzy extractor. Then, U_i compute the

$$M_{R1} = (ID_i \parallel PW_i \parallel b_i \parallel \sigma_i),$$

$$Reg_i = h(b_i \parallel \sigma_i),$$

and send$\{M_{R1}, Reg_i, TS_{R1}\}$ to the GWN.

Step 2: Upon receiving a message $\{M_{R1}, Reg_i, TS_{R1}\}$ from U_i, GWN verifies the validity of TS_{R1} by compares with the extracted local timestamp TS'_{R1}. If the dissimilarity exceeds the threshold value known as maximum transmission delay ΔT, the request is terminated. the request is terminated, CS checks for (ID_i), if it existed in its database or not. If it exited, the CS computes

$$TC_i = \mathcal{H}(K_{GWN-U} \parallel ID_i \parallel TE_i),$$

$$PTC_i = TC_i \oplus \bar{\mathrm{h}}(M_{R1} \parallel \sigma_i^*),$$

and store (ID_i, TE_i) in the proof table. Then, GWN fills the smartcard with (H (), Y, TE_i, PTC_i), and forwards it to U_i.

Step 3: Upon receiving parameters from GWN, U_i starts verifies the timestamp value $TS'_{R2} - TS_{R2} < \Delta$ T. If verification is successful, U_i computes

$$M'_{R1} = \mathrm{H}\ (\mathrm{h}\ (\ ID_i \parallel \mathrm{P}W_i \parallel \sigma_i \parallel b_i)\ \mathrm{mod}\ \mathrm{m})$$

where m is a medium integer, $2^8 \leq m \leq 2^{16}$,which regu lates the volume of < IDi, PWi > pair contrary tod an offline-password-guessing outbreak.

Again computes

$$TC_i = PTC_i \oplus \mathrm{h}(M_{R1} \| \sigma_i)$$

and stores TC_i, b_i, M'_{R1} in the User's memory. In the end, the smartcard holds the $(H(.)$,Y, $TE_i, PTC_i, b_i, M'_{R1})$.

Step 4: Similarly, sensor node S_j registers with GWN by sending SID_j through a secure channel. Then GWN computes

$$TC_j = \mathrm{H}\left(K_{GWN-S} \parallel SID_j\right)$$

and forwards TC_j as credentials to the sensor node, which stores it in its secured database Fig.5.

4.3.2 Login and Authentication Phase

Step 1: When User (U_i) needs to communicate or access Sensor node (S_j), U_i inserts his/her smart card into the terminal and inputs user identity ID_i, password PW_i, and Biometric info Bio'. Then smartcard computes

$$\sigma_i^* = \mathrm{Rep}(\tau_i, \mathrm{Bio}),$$

$$M_{R1}^* = \mathrm{H}(\mathrm{h}((ID_i \parallel PW_i \parallel \sigma_i \parallel b_i) \bmod \mathrm{m})$$

and verifies $M_{R1}^* \stackrel{?}{=} M_{R1}$. If the calculation is unsafe, the smartcard drops the message and aborts further operations. Otherwise, it computes

$$TC_i = PTC_i \oplus (\mathrm{H}\ (ID_i \parallel PW_i \parallel b_i.$$

Upon successful login, the mutual authentication process with a shared session key implemented (Fig.6). User U_i chooses two random numbers a ϵZ_{p-1}^*; pick another random number as r_i and starts computations of

$$G_1 = \mathrm{AP}\ G_2 = \mathrm{AY} = \mathrm{AXP};$$

$$DID_i = ID_i \oplus (H(G_1 \parallel G_2),$$

$$\mathrm{Regi} = \mathrm{H}(b \parallel \sigma_i),$$

$$A=h\ (Reg_i \parallel DID_i \parallel TC_i \parallel TS_{A2}),$$

$$M_{A1} = H)\ ID_i \parallel TC_i \parallel G_1 \parallel G_2 \parallel r_i).$$

Finally, U_i forwards (M_{A1}, Reg_i, TS_{A2}) to the GWN for further calculation with required authentication information.

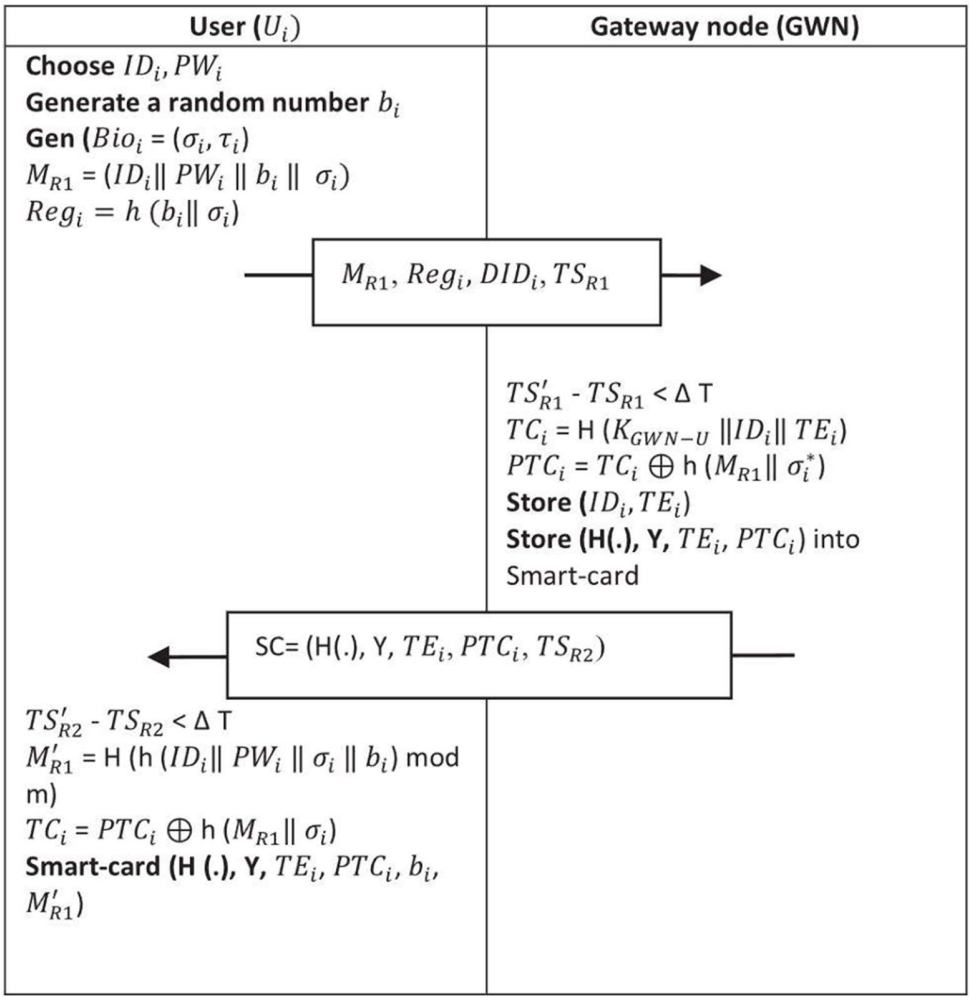

Fig. 5. Registration Phase of the *Chander et al.'s* Scheme.

Step 2: Upon receiving the forwarded message from U_i, GWN verifies the freshness of the message by checking if $TS'_{A2} - TS_{A2} < \Delta$ T. If the difference exceeds the threshold value known as maximum transmission

delay ΔT, GWN terminates the request message. Otherwise, GWN again checks

$$A_I' = \mathcal{H}(Reg_i \,||\, DID_i \,||\, TC_i \,||\, TS_{A2}),$$

then generates $G_2 = xY$ =xaP; It then computes the value of

$$ID_i = DID_i \oplus (\mathcal{H}(G_1 \,||\, G_2),$$

$$TC_i = \mathcal{H}(K_{GWN-U} \,||\, ID_i \,||\, TE_i)$$

and checks if H ($ID_i \,||\, TC_i \,\|\, G_1 \,||\, G_2 \,||\, r_i \,||\, TS_{A1}) \stackrel{?}{\equiv} M_{A1}$,checks $M_{A1}{}' \stackrel{?}{\equiv} M_{A1}$. If these values do not match, GWN discards the application; otherwise, it chooses a sensor node S_j, generates a random number r_g, and computes

$$TC_j = \text{H}\,((K_{GWN-U} \,||\, ID_i| \,|\, TE_i) \oplus r_g,$$

$$DID_GWN = ID_i \oplus (\text{H}\,\big(DID_I \,||\, TC_j \,||\, TS_{A2} \,||\, r_g\big).$$

After that, GWN extracts the local timestamp value to compute

$$M_{G1} = \text{H}\big(ID_i \,||\, TC_j \,||\, TS_{A2} \,||\, G_2\big).$$

Finally, GWN forwards $(DiD_i, DID_{GWN}, M_{G2}, G_1, TS_{A3})$ to the sensor node S_j.

Step 3: Upon receiving the message from GWN, the sensor node S_j verifies the freshness of the message by checking if $TS_{A3}' - TS_{A3}' < \Delta$ T, if the calculation of TS is invalid, S_j drops the message, otherwise, $S_j$computes

$$ID_i = DID_{GWN} \oplus \text{H}\big(DID_I \,||\, TC_j \,||\, TS_{A2} \,||\, r_g\big)$$

and checks if $M_{G1} \stackrel{?}{\equiv} \text{H}\big(ID_i \,||\, TC_j \,||\, TS_{A2} \,||\, G_2\big)$. If calculations are equal, S_j generate a random number b$\varepsilon Z_{p-1}{}^*$, and compute

$$G_j = \text{bp};$$

$$SK_ij = \text{H}\,(\text{b}G_1) = \text{H}\,(\text{abp});$$

$$M_{S3} = H\big(ID_i \,||\, TC_j\big|| \,SID_j \,||\, G_j\big||TS_{A4}\big).$$

Finally, forwards $(SID_j, G_J, M_{S3}, TS_{A4})$to GWN.

Step 4: GWN checks the validity by comparing it with the extracted local timestamp TS'_{A4}. If the dissimilarity is greater than the threshold value ΔT, the request message is terminated by GWN. If it was lesser than the threshold value, GWN computes

$$M_{S3} \stackrel{?}{\equiv} \mathrm{H}(ID_i \parallel TC_j \parallel SID_j \parallel G_j \parallel TS_{A4})$$

equal or not. If it is equal, GWN confirms that the sensor node S_j is authenticated then generates E_{GWN} =H($ID_i \parallel TC_i \parallel' SID_j \parallel G_2 \parallel G_j \parallel TS_{A5}$) and forwards ($SID_j, TS_{A5}, G_j, E_{GWN}$) to U_i.

Step 5: Upon receiving the message from GWN, U_i check the freshness of the received message by verifying if $TS'_{A5} - TS_{A5} < \Delta$ T. If freshness is within the threshold value, U_i verify the $E_{GWN} \stackrel{?}{\equiv}H(ID_i \parallel TC_i \parallel SID_j \parallel G_2 \parallel G_j \parallel TS_{A5})$. If verification is successful, U_i conforms that S_j, GWN is authenticated. Lately, U_i computes the shared session ker

$$SK_{ij} = \mathcal{H}(bG_j) = \mathcal{H}(\text{abp}).$$

4.3.3 Password Change Phase

Step 1: In this phase, the User can change or update the password without the need for communication with GWN or the sensor node.

After taking input from U_i and capturing the biometric information using the fuzzy extractor, if the user desires to modify the password, the smart card computes

$$\sigma_i^* = \mathrm{Rep}(\tau_i, \mathrm{Bio}),$$

$$M_{R1}^* = \mathrm{H}\ (\mathrm{h}((ID_i \parallel PW_i \parallel \sigma_i \parallel b_i) \mathrm{mod\ m})$$

if the $M_{R1}^* \stackrel{?}{\equiv} M_{R1}$ is valid, U_i chooses the new password PW_i', and computes

$$PTC_i' = TC_i \oplus \mathrm{h}\ (M_{R1} \parallel \sigma_i^* \parallel PW_i').$$

Here PTC_i replaces with PTC_i'.

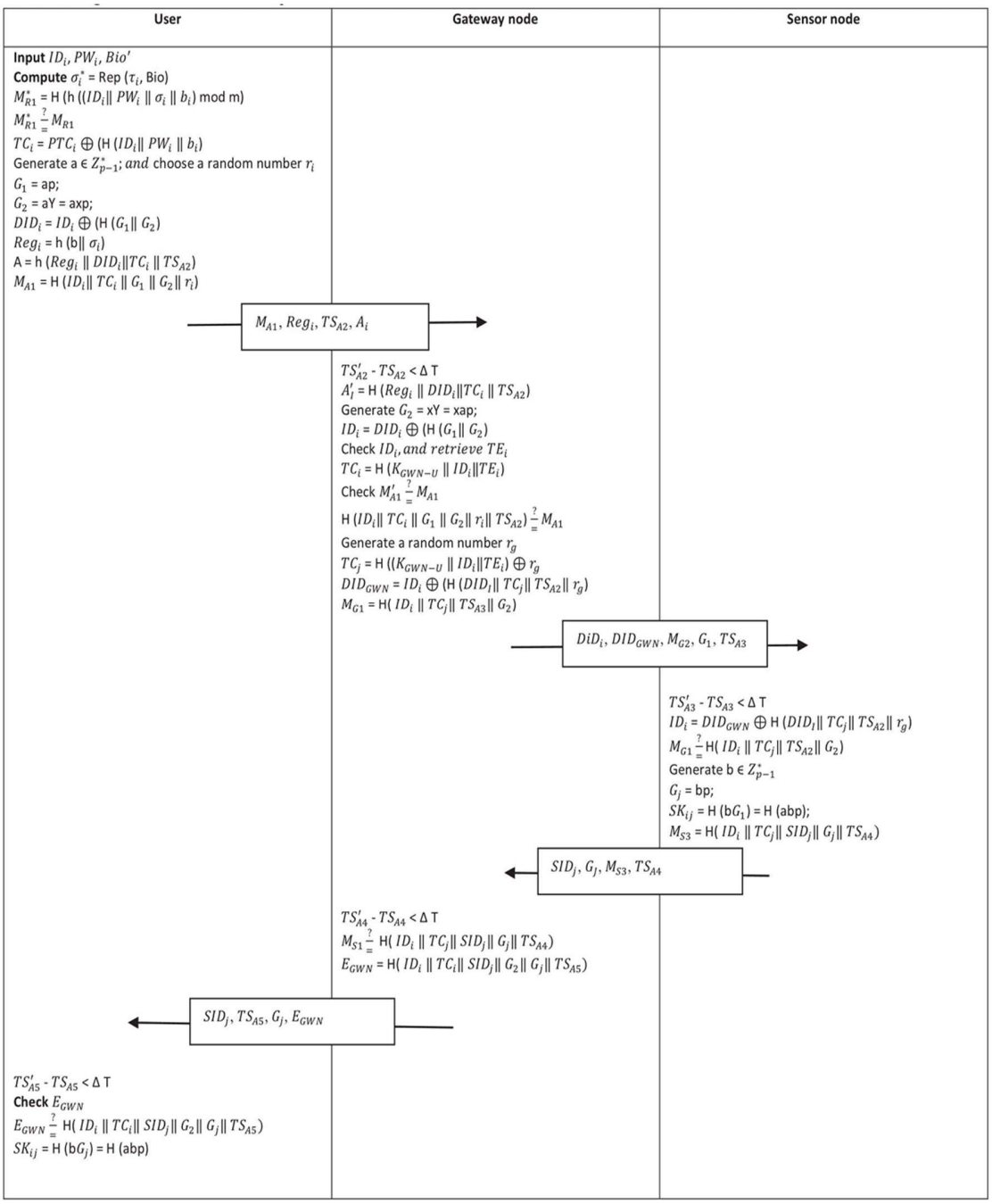

Fig. 6. Login and Authentication Phase of the *Chander et al.'s* Scheme.

4.4 Common issues and suggestions

The above 3 literatures all share some common problems. First, the complexity of resource constrained environments: all three schemes aim to provide security for wireless sensor networks (WSNS), which are typically resource constrained. The complexity of cryptographic operations may be too demanding for such environments. Secondly, there are some scalability issues, and these schemes may need to be scalable to accommodate a large number of devices in WSNS, which is not explicitly addressed in all papers. Then, while all papers claim to resist various attacks, the specific details of how they resist emerging threats or advanced attacks are not fully elaborated. Finally, there may be a trade-off between security and ease of use. Complex security measures may make the system more difficult for the end user to interact with.

In view of the above problems, this course work puts forward the following suggestions:

- Simplify encryption operations: Optimize the encryption algorithm to reduce the computational load on the sensor nodes without sacrificing security. For example,
  a) A more efficient key agreement protocol is adopted to reduce the calculation and communication overhead in the process of key exchange.
  b) Use hardware accelerators or dedicated chips to perform encryption operations, reducing the burden on the main processor.
- Enhanced scalability: The authentication scheme is designed with scalability in mind in order to efficiently handle the increasing number of devices. For example,
  a) The layered authentication architecture is introduced so that local authentication problems will not affect the whole network.
  b) Adopting a modular design scheme, it is convenient to add new security features as needed in the future.
- Enhanced ease of use:
  a) Intuitive user interface and operation flow are designed to reduce user learning costs.
  b) Provides customized security options that allow users to adjust security Settings according to their needs and risk appetite.
- Security enhancements:

a) Security protocols are regularly reviewed and updated to address emerging security threats.
b) Introduce advanced techniques such as machine learning to predict and defend against unknown attack patterns.

- User-centric design: strikes a balance between security and user experience, ensuring that the system is not only secure, but also user-friendly. For example,
  a) Develop user guidelines and best practices that are easy to understand and operate to increase user awareness and acceptance of security features.
  b) Provide multilingual support and culturally adaptable design to meet the needs of users in different regions.
- Periodic security updates: Implement mechanisms to periodically update security protocols to protect the system against new threats. For example,
  a) Implement automated security scanning and vulnerability detection mechanisms to detect and fix security vulnerabilities in a timely manner.
  b) Establish a quick response team to respond to and deal with security incidents in a timely manner.
- Performance Evaluation: A comprehensive performance evaluation, including real-world tests, is conducted to ensure that the scheme performs well under a variety of conditions. For example,
  a) Long-term performance monitoring and evaluation are carried out to ensure the effectiveness and stability of security solutions in practical applications.
  b) Performance evaluations, including user feedback, ensure that security solutions do not negatively impact the user experience.

Among the above three security schemes, Li et al. 's scheme is relatively the best. Because through the security and performance comparison (Table 2, Table 3), Li et al. 's scheme provides the best security against all known attacks. Moreover, although Li et al. 's scheme requires higher computation time compared to Chander et al.' s scheme, it requires less communication overhead. Therefore, Li et al. 's scheme achieves the best balance among security, computational efficiency and communication cost, and is relatively the best scheme.

Chapter 5

COMPARATIVE ANALYSIS

This section compares Hu et al. 's scheme [3], Li et al.' s scheme [5], and Chander et al. 's scheme [4] with other schemes of the same type in terms of computational performance and security.

5.1 Comparison of Safety Features and Functions

As you can see from Table 2, the three most recent two-factor authentication schemes have higher security than the others, and are able to resist most or all known attacks. From the perspective of a single scenario, the 2 proposed by Li et al. and Chander et al. offers the most comprehensive security advantage against all known attacks.

5.2 Computation performance and Communication cost comparison

This section compares the computational efficiency of the scheme presented in this course work with existing methods and lists the results in Table 2. Since registration is performed only once, the comparative computational efficiency of login and authentication is estimated. Since these two operations play an important role in running the expected authentication scheme. This course work conveniently ignores the calculation of the XOR operation because it is negligible. In addition, only four major encryption operations are considered in this course work: (1) one-way hash functions, (2) point multiplication, (3) symmetric encryption and decryption, and (4) fuzzy extraction functions. The encryption times of the Hash function, encryption/decryption, point multiplication in elliptic curves, and fuzzy extraction function are denoted as T_h, T_s, T_e, and T_f, and the estimated time are 0.068, 0.56, 2.501, and 2.501 ms, respectively. As can be seen from the results in Table 3, the comparison results of the three schemes introduced in this course work are similar. The scheme proposed by Chander et al. has the best computational efficiency, while the scheme proposed by Li et al. has the lowest communication cost. In summary, the scheme proposed by Li et al. has more advantages. Compared with other similar schemes, it can be seen that these three schemes achieve the best balance between computational efficiency and communication cost.

Scheme	*R1*	*R2*	*R3*	*R4*	*R5*	*R6*	*R7*	*R8*	*R9*	*R10*	*R11*
Hu *et al*	√	√	√	√	√	×	√	√	×	√	√
Li *et al*	√	√	√	√	√	√	√	√	√	√	√
Chander *et al*	√	√	√	√	√	√	√	√	√	√	×
Ref [17]	×	√	√	×	√	√	√	√	√	√	√
Ref [18]	√	√	×	√	×	×	×	√	√	√	√
Ref [19]	×	√	×	×	×	×	×	√	×	√	×
Ref [20]	√	×	×	√	×	√	×	×	√	×	√

TABLE.2 Security comparison among relevant schemes

【R1: User anonymity; R2: User untrace ability; R3: Mutual authentication; R4: Resistance to reply attack; R5: Resistance to MITM attack; R6: Resistance to DOS attack; R7: Forward security; R8: Resistance to impersonation attack; R9: Resistance to stolen smart card attack; R10: Resistance to known session key attack; R11: Resistance to off-line password guessing attack; √ denotes the scheme can provide the corresponding attribute; × denotes the scheme cannot provide the corresponding attribute】

Scheme	*User*	*GWN*	*Sensor node*	*Total*	*Time (ms)*	*Communication cost (Bits)*
Hu *et al.*	$7T_h+3T_e$	$10T_h+T_e$	$6T_h+2T_e$	$23T_h+6T_e$	16.570	4981
Li *et al.*	$11T_h+3T_e$	$14T_h+T_e$	$7T_h+2T_e$	$32T_h+6T_e$	17.182	4540
Chander *et al.*	$6T_h+2T_e$	$8T_h+T_e$	$5T_h+T_e$	$19T_h+4T_e$	14.382	5120
Ref [17]	$16T_h+5T_e+T_f$	$9T_h+3T_e$	$8T_h+4T_e$	$33T_h+12T_e+T_f$	34.757	3771
Ref [18]	$8T_h+3T_e+T_s$	$7T_h+T_e+2T_s$	$5T_h+2T_e+T_s$	$20T_h+6T_e+4T_s$	18.606	4451
Ref [19]	$17T_h+6T_e+T_f$	$12T_h+3T_e$	$8T_h+4T_e$	$37T_h+13T_e+T_f$	37.530	3874
Ref [20]	$10T_h+4T_e+2T_s+T_f$	$5T_h+2T_e+T_s$	$5T_h+2T_e+2T_s$	$20T_h+8T_e+5T_s+T_f$	26.669	5671

TABLE.3 Computational and Communication cost of the schemes

Chapter 6

CONCLUSION

Although with the booming development of Internet of Things technology, more and more literature began to pay attention to WSN and related security research. However, at the same time, the means of attack against WSN are constantly evolving and updating. The contradiction between increasingly stringent security requirements and the limited performance of physical devices in WSN makes the design of secure and efficient attack defense mechanisms more complex. The two-factor authentication scheme for WSN has been paid more and more attention by researchers in recent years because of its better security and excellent computing efficiency. The implementation principle of two-factor authentication schemes is mostly based on ECC encryption algorithm, but because of the different security, the performance difference between different schemes is very different. Therefore, analyzing the actual performance of the latest WSN two-factor authentication scheme can help readers quickly understand the relevant research and characteristics in recent years.

This course work summarizes 3 excellent WSN two-factor authentication schemes in recent years and introduces them mathematically. Then, we compare them with the existing schemes in terms of security and computational performance and give the results. The comparison results show that the two-factor authentication schemes can resist most of the known attacks and achieve a good balance with the computing performance. In addition, with the continuous improvement of researchers, the two-factor authentication scheme not only has better security, but also further reduces the computational cost and further increases the computational efficiency. The latest security scheme proposed by Li et al. can not only meet the limited performance of physical devices in WSN, but also resist all known attacks.

In the future, with the continuous update of attack methods, the security requirements of authentication schemes will become more and more stringent. Therefore, how to maintain or even achieve better computational efficiency while meeting higher security requirements will require more literature research and exploration. By adopting actions such

as optimizing encryption algorithms (such as introducing lightweight algorithms other than ECC), it may be possible to further enhance security while improving computational performance to match the limited device performance in WSN.

References

[1] Tran-Dang H, Krommenacker N, Charpentier P, et al. "Toward the Internet of Things for physical Internet: Perspectives and challenges." *IEEE Internet of Things Journal 7*, no. 6 (2020): 4711-4736.

[2] Bin Abu Bakar K, Zuhra F T, Isyaku B, et al. "A review on the immediate advancement of the Internet of Things in wireless telecommunications." *IEEE Access* 11 (2023): 21020-21048.

[3] Hu B, Tang W, Xie Q. "A two-factor security authentication scheme for wireless sensor networks in IoT environments." *Neurocomputing* 500 (2022): 741-749. doi: 10.1016/j.neucom.2022.05.099.

[4] Chander B, Kumaravelan G. "An Improved 2-Factor Authentication Scheme for WSN Based on ECC." *IETE Technical Review* 40, no. 2 (2023): 167–178. doi: 10.1080/02564602.2022.2055671.

[5] LI A, KANG B, ZUO X, HUO Y, NIU S, SUN Z. "Analysis and Improvement on an Authentication Scheme for Wireless Sensor Networks in Internet of Things Environment." *Wuhan Univ. J. Nat. Sci*. 28, no. 6 (2023): 541-552. doi: 10.1051/wujns/2023286541.

[6] Du J Q, Kang B Y, Han Y B. "Improvement on a biometric based user authentication scheme in wireless sensor networks using smart cards." *Wuhan University Journal of Natural Sciences* 25, no. 2 (2020): 155-161.

[7] Szymoniak S, Kesar S. "Key agreement and authentication protocols in the Internet of Things: A survey." *Applied Sciences* 13, no. 1 (2022): 404.

[8] Abbas S, Al-Abrrow H, Abdullah HO, et al. "Encountering Covid-19 and perceived stress and the role of a health climate among medical workers." *Current Psychology* 41, no. 12 (2022): 9109–9122.

[9] Lamport L. "Password authentication with insecure communication." *Communications of the ACM* 24, no. 11 (1981): 770-772.

[10] Kesavan V T, Radhakrishnan S. "Cluster based dynamic keying technique for authentication in wireless sensor networks." *Communications in Computer and Information Science* 296 (2013): 1-8.

[11] Jiang C, Bao L, Xu H. "An efficient scheme for user authentication in wireless sensor networks." *In International Conference on Advanced Information Networking & Applications Workshops*, pp. 438-442 (2007).

[12] Sharif A O, Arshad H, Nikooghadam M, Abbasinezhad-Mood D. "Three party secure data transmission in IoT networks through design of a lightweight authenticated key agreement scheme." *Future Generation Computer Systems* 100, no. 1 (2019): 882-892.

[13] Chen C T, Lee C C, Lin I C. "Efficient and secure three-party mutual authentication key agreement scheme for WSNs in IoT environments." *PLoS ONE* 15, no. 4 (2020): e0232277.

[14] Das M L. "Two-factor user authentication in wireless sensor networks." *IEEE Transactions on Wireless Communications* 8, no. 3 (2009): 1086-1090.

[15] Khan M K, Alghathbar K. "Cryptanalysis and security improvements of 'Two-factor user authentication in wireless sensor networks'." *Sensors* 10, no. 3 (2010): 2450-2459.

[16] Vaidya B, Makrakis D, Mouftah H. "Two-factor mutual authentication with key agreement in wireless sensor networks." *Security Communications Networks* 9, no. 2 (2016): 171-183.

[17] Sutrala A K, Obaidat M S, Saha S, et al. "Authenticated key agreement scheme with user anonymity and untraceability for 5G-enabled softwarized industrial cyber-physical systems." *IEEE Transactions on Intelligent Transportation Systems* 23, no. 3 (2022): 2316-2330.

[18] Xie Q, Li K H, Tan X, et al. "A secure and privacy-preserving authentication protocol for wireless sensor networks in smart city." *EURASIP Journal on Wireless Communications and Networking* (2021): 119.

[19] Srinivis J, Das A K, Wazid M, et al. "Designing secure user authentication protocol for big data collection in IoT-based intelligent transportation system." *IEEE Internet of Things Journal* 8, no. 9 (2021): 7727-7744.

[20] Sahoo S S, Mohanty S, Majhi B. "A secure three factor based authentication scheme for health care systems using IoT enabled devices." *Journal of Ambient Intelligence and Humanized Computing* 12, no. 1 (2021): 1419-1434.