scientific reports

Check for updates

OPEN

ECC-based three-factor authentication and key agreement scheme for wireless sensor networks

Wenfeng Huang

In wireless sensor networks (WSNs), protocols with authentication and key agreement functions can enhance the security of the interaction between users and sensor nodes, guaranteeing the security of user access and sensor node information. Existing schemes have various security vulnerabilities and are susceptible to security attacks (e.g., masquerading user, password guessing, internal privilege, and MITT attacks), so they cannot meet the anonymity requirements or achieve forward security. To effectively improve the security performance of WSNs, an elliptic curve cryptography (ECC)-based three-factor authentication and key agreement scheme for WSNs is proposed. The scheme is based on the ECC protocol and combines biometrics, smart card and password authentication technology; uses a challenge/response mechanism to complete the authentication between users, gateways, and sensors; and negotiates a secure session key. The Burrows, Abadi and Needham logic for formal security analysis proves the correctness and security of the scheme, and the informal analysis of multiple known attacks proves that the scheme can resist various attacks and has high security characteristics. The feasibility of the scheme has been analysed and verified with the ProVerif tool. The efficiency analysis results show that the scheme is suitable for resource-constrained WSNs.

As wireless sensor networks (WSNs) are widely used in various application areas, securing their communication has become one of the focuses of researchers. The confidentiality of information communication is a major challenge, and protecting the privacy of data from unauthorized access by attackers is a major problem facing Internet of Things (IoT) WSNs[1]. Current schemes suffer from various security vulnerabilities in authentication and key agreement functions and are susceptible to security attacks such as masquerading users, password guessing, insider privileges, and MITM (Man-in-the-Middle), so they cannot satisfy anonymity requirements or achieve forward security. In IoT WSNs, establishing user authentication protocols with session keys is an approach that is widely used to solve the above problems. In this context, this study aims to address the security vulnerabilities in existing WSNs, especially in the interaction between users and sensor nodes, to ensure the security of user access and sensor node information.

The significance of this research lies in the following points: (1) Safeguarding communication security: WSNs are widely used in environmental monitoring, health care, intelligent transportation, etc., which include data communication that often involves personal privacy and important information. By improving the security of authentication and key agreement, this study helps to secure user access and sensor node information against potential attack risks. (2) Filling existing security holes: In this study, it is found that there are various vulnerabilities in the current security protocols in WSNs, which may be subject to attacks such as camouflage and password guessing. By combining elliptic curve cryptography and multifactor authentication techniques, this scheme is expected to fill these loopholes and improve the overall security of WSNs. (3) Promotion of the development of security in the field of WSNs: With the evolution of the IoT, the range of applications of WSNs is expanding. Research on communication schemes with high security is crucial for the healthy development of WSNs. This study aims to offer fresh insights and approaches for enhancing security in WSNs. (4) Positive impact on practical applications: Not only is the correctness and security of the scheme verified through formal BAN logic and the ProVerif tool, but its ability to fight against a wide range of attacks through informal analysis is also verified. This makes the scheme more likely to succeed in practical applications and provides strong technical support for real-world deployments. (5) Suitable for resource-constrained environments: The results of the efficiency analysis

School of Information Engineering, Xiamen Ocean Vocational College, Xiamen 361100, Fujian, China. email: hwfjaxx@163.com

show that the scheme is suitable for resource-constrained WSNs. This is a substantial advantage for sensor nodes that have limited computational and storage resources and is expected to have a positive impact in the real world.

To effectively enhance the security performance of WSNs, this study proposes a three-factor authentication and key agreement scheme based on elliptic curve cryptography (ECC). The scheme is based on the ECC protocol, combines biometric, smart card and cryptographic authentication techniques, uses a challenge/response mechanism to complete the authentication between the user, the gateway and the sensor, and negotiates a secure session key. The correctness and security of the scheme are validated through formal security analysis using BAN logic. In addition, the scheme is verified as highly secure against various attacks through informal analysis of a variety of known attacks. To ensure the feasibility of the research, the paper also provides an exhaustive analysis and validation of the scheme using the ProVerif tool. The final efficiency analysis results show that the scheme is suitable for resource-constrained WSNs and provides a feasible and efficient solution for secure communication in WSNs. The purpose of this study is to promote the development of security in the field of WSNs and to provide a more reliable protection mechanism for wireless sensor networks in practical applications.

Related works

In 2015, Lee et al.[2] proposed a nontamper smart card authentication key protocol scheme based on anonymous passwords. In 2017, Wu et al.[3] noted that the scheme of Lee et al.[2] is not resistant to smart card loss, spoofed users, spoofed server attacks, and so forth. Wu et al. proposed an enhanced anonymous password authentication key agreement scheme. In 2016, Jiang et al.[4] proposed a two-factor authentication scheme based on ellipse curve cryptography (ECC) for untraceable time vouchers in WSNs. In 2018, Li et al.[5] found flaws in the work of Jiang et al.[4], such as the lack of a password detection and change mechanism and a clock synchronization problem. Thus, Li et al. proposed a three-factor anonymous authentication scheme for WSNs in the IoT environment, using a fuzzy commitment scheme and error correction code to process user biometric information; however, the scheme proved to be unable to resist smart card loss attacks and achieve forward security. In 2022, Meriam et al.[6] performed an informal security analysis of the protocol of Li et al.[5], and the results showed that it cannot achieve anonymity and cannot resist session key leakage, internal, and other attacks. Thus, Meriam et al. proposed a three-factor mutual authentication and key agreement protocol for IoT WSNs based on lightweight ECC, using physically unclonable functions (PUFs) and ECC to improve security and effectively solve the security problem of Li et al.'s proposal[5].

In 2017, Wu et al.[7] proposed a user authentication scheme for WSNs based on the Internet of Things(IoT) and, in the same year, an efficient authentication and key agreement scheme for multigateway WSNs in the deployment of the IoT[8]. In 2019, Bayat et al.[9] noted that the scheme of Wu et al.[7] could not withstand certain security attacks. Thus, Bayat et al. proposed an analysis and improvement of the user authentication scheme of the IoT based on ECC. In 2019, Guo et al.[10] found that the scheme of Wu et al.[8] was inefficient and instead proposed a secure and efficient three-factor multigateway authentication protocol for WSNs; however, this scheme proved to be unable to resist offline password guessing and other attacks. In 2017, Jung et al.[11] proposed an efficient and secure anonymous authentication scheme based on key agreement in WSNs. In the same year, Sravani et al.[12] proposed an authentication key establishment scheme based on a secure signature for future IoT applications. However, the scheme was not resistant to man-in-the-middle attacks and was too complex and inefficient[13].

In 2021, Azrour et al.[14] proposed a new, enhanced IoT authentication protocol based on the literature[2,5], and[9], that could resist replay, internal, and other attacks. In 2021, Vinoth et al.[15] proposed a multifactor authentication key protocol scheme for industrial IoT security; however, this scheme could not deal with certain types of attacks, such as sensor node capture and replay attacks. In 2021, Xue et al.[16] proposed a lightweight three-factor authentication and key agreement scheme for multigateway WSNs in the IoT based on a ummary of the literature[10,14], and [15] and proved the correctness and security of the proposed scheme through the BAN logic and BPR model. However, the scheme could not guarantee the security of the user's private key or negotiate a secure session key.

Motivation, contributions and road-map

Motivation

The motivation of this paper is to improve the security of wireless sensor networks (WSNs), especially to enhance the authentication and key agreement features in the interaction between users and sensor nodes. Currently existing schemes suffer from various security vulnerabilities and are susceptible to security attacks such as masquerading users, password guessing, internal privileges, and man-in-the-middle attacks. These vulnerabilities make it difficult for existing schemes to meet anonymity requirements and achieve forward security. In this article, they propose an integrated authentication and key agreement scheme based on the ECC protocol is proposed, combining multiple authentication techniques to improve the security performance of WSNs, and demonstrate its feasibility and high level of security through formal and informal security analysis.

Their contribution

1) This paper proposes a three-factor authentication and key agreement scheme based on ECC for WSNs[17]. The new scheme is based on the ECC key agreement mechanism and introduces the challenge/response mechanism to establish authentication and key agreement mechanisms among users and gateways and sensors of WSNs. The security of the scheme is guaranteed by the security characteristics of biometrics, the elliptic curve discrete logarithm problem, and the one-way characteristics of the hash function.

2) After the authentication and key agreement between the user and the sensor is completed, a password update and smart card logout scheme is proposed to assist users in better managing smart cards and enhance the security of the scheme.
3) The proposed scheme is validated in several forms. The scheme's security is assessed through a formal analysis employing BAN logic. In addition, the nonformal security analysis proves the security performance of the scheme and its resistance to various attacks. Furthermore, simulations using the ProVerif tool validate the feasibility of the proposed scheme. Finally, the performance analysis shows that the scheme improves security without increasing energy consumption.

The road-map of the paper is as follows
In Section "Mathematical preliminaries", they reviewed some of the basics of math and information security and defined the notations and descriptions and threat model used by the scheme. In Section "Safety analysis of existing schemes", the advantages and some security vulnerabilities in the work of Xue et al.[16] are discussed. Sections "The proposed scheme" and "Security analysis" present the proposed scheme and the corresponding security analysis, respectively. In Section "Efficiency analysis", the performance of the proposed scheme is evaluated, and finally, the whole paper is concluded in Section "Conclusions".

Mathematical preliminaries

Cryptanalysis

Cryptanalysis, a subset of cryptography, is the process of deciphering or breaking cryptographic systems. It utilizes techniques such as mathematics, computer science, and engineering to unveil encrypted data. The primary objective of cryptanalysis is to achieve unauthorized access to encrypted information by scrutinizing weaknesses in encryption algorithms, key management, and security mechanisms. This involves activities such as password guessing, analysing the mathematical aspects of encryption algorithms, identifying vulnerabilities in encryption keys, and exploiting errors in implementation. The efficacy of cryptanalysis hinges on the intricacy and robustness of the cryptosystem. This field plays a pivotal role in information security, contributing to the evaluation and enhancement of cryptographic system strength.

ECC and ECDH[18]

Elliptic Curve Cryptography (ECC) is a public key encryption algorithm that is widely used in the field of cryptography. The security of ECC is based on the discrete logarithmic problem on elliptic curves, which is considered to be difficult to solve; thus, encryption algorithms based on this mathematical puzzle provide a high level of security. Compared to traditional RSA algorithms based on the integer factorization problem, ECC can use shorter key lengths while providing the same level of security, thus reducing the computational and storage requirements. Overall, elliptic curve cryptography is an important part of the modern field of cryptography and provides a powerful tool for secure communication.

The elliptic Curve Diffie-Hellman key exchange (ECDH) is mainly used to establish secure shared encryption data in an insecure channel, generally exchanging private keys, which are generally used as "symmetric encryption" keys by both parties for subsequent data transmission. ECDH is based on the premise that given a point P on an elliptic curve and an integer k, it is easy to solve for Q = KP, but it is difficult to solve for K via Q, P.

BAN logic

BAN logic is a formal method for analysing and verifying cryptographic schemes, proposed by Burrows, Abadi, and Needham (BAN) in 1989[19]. The basic idea of BAN logic is to convert messages in a cryptographic scheme into a logical language representation and then use inference rules to derive the beliefs and goals of the participants in the scheme. BAN logic can be used to find vulnerabilities in a scheme to improve its security and efficiency.

Table 1 shows the notations used by BAN logic[20] and descriptions of these notations. The BAN logic rules used include: message meaning rule R1: $\frac{P|\equiv P \overset{SK}{\leftrightarrow} Q, P \triangleleft \{H\}_{SK}}{P|\equiv Q|\sim H}$, random number verification rule R2: $\frac{P|\equiv \#(H), P|\equiv Q|\sim H}{P|\equiv Q|\equiv H}$, arbitration rule R3: $\frac{P|\equiv Q|\equiv H, P|\equiv Q|\Rightarrow H}{P|\equiv H}$, freshness rule R4: $\frac{P|\equiv \#(H)}{P|\equiv \#(H,G)}$, belief rule R5: $\frac{P|\equiv (H,G)}{P|\equiv G}$, and session secret key rule R6: $\frac{P|\equiv \#(H), P|\equiv Q|\equiv H}{P|\equiv P \overset{SK}{\leftrightarrow} Q}$.

Random oracle model

In 1993, Bellare and Rogaway formally proposed the Random Oracle Model (ROM) methodology, with which the past purely theoretical research of provable security methodology quickly made significant progress in the

Notations	Descriptions	Notations	Descriptions
$P\mid\equiv H$	P believes H is true	$P \overset{SK}{\leftrightarrow} Q$	Both P and Q can use the shared key SK to communicate with each other
$P \triangleleft H$	P sees H and is capable of reading and repeating it	$P\mid\sim H$	P once said H; at some time, P has sent the message containing H
$\#(H)$	H is fresh which means it was never sent before the current execution of the protocol	$P\mid\Rightarrow H$	P has control or jurisdiction over H
$\{H\}_K$	The ciphertext obtained by encrypting plaintext H with key K		

Table 1. Notations used by BAN logic and descriptions of these notations.

field of practical applications. A large number of fast and effective security programs have been proposed, and at the same time, they also produced the "concrete security or exact security", which means that they no longer only satisfy the asymptotic degree of security but can exactly obtain a more accurate security measure. Practical-oriented provable security theory has been widely accepted by academia and industry.

Inside cryptography, a random oracle is a prediction machine (simply put, like a black box for the theory) that returns a truly uniformly random output for any input, and for the same input, this prediction machine outputs the same output in the same way every time (i.e., if the query is repeated, it responds in the same way every time the query is submitted). In other words, a randomized prediction machine is a function that randomly maps all possible inputs to outputs.

The stochastic prediction machine model is usually an idealized stand-in for the real hash function and has its origins in the idea of viewing hash functions as pseudorandom. The stochastic prediction machine model has the following properties:

1) Consistency: Inputs that are the same should produce matching outputs.
2) Computability: the output can be calculated within a polynomial time frame.
3) Uniform Distributability: The prediction machine's output is evenly spread across the value space without any overlaps.
4) In the stochastic prediction machine model, it is assumed that the adversary will not exploit the weakness of the hash function to attack the cryptographic scheme.

Notations and descriptions

Table 2 shows the notations used in this paper and descriptions of these notations.

Threat model[18]

In this article, the following threat models are used:

1) Communication conducted over a public channel is susceptible to eavesdropping, providing attackers with an advantage.
2) Threats to any system can come from external entities or even legitimate users who may act as attackers.
3) Attackers have the capability to manipulate, erase, redirect, and replay intercepted messages, compromising the integrity of the communication.
4) The attacker is assumed to possess knowledge of the protocol used in the authentication system.

Safety analysis of existing schemes[16]

Scheme[16] proposed an authentication and key agreement scheme for multigateway environments. In the scheme, biometrics, a crucial element, is extracted and authenticated using a fuzzy extractor. The program consists of the following six processes:

1) System initialization. The *SA* assigns identity ID_{hg}, ID_{fg} and private keys x_{hg}, x_{fg} to *HGWN* and *FGWN* and establishes a shared key K_{hf}. The *HGWN* and *FGWN* independently choose three random numbers, denoted as R_h, R_f and R_{fh}, respectively.

Notations	Descriptions	Notations	Descriptions
U_i and U_a	User U_i and U_a	$E(F_p)$	Elliptic curve finite field $E(F_p)$
GWN	Gateway node GWN	P	Base points on elliptic curves P
S_j	Sensor node S_j	r_i and r_u	The private key r_i and r_u of the user U_i
ID_i	The identity ID_i of the user U_i	r_g	The private key r_g of the gateway node GWN
ID_{hg}	The identity ID_{hg} of the gateway node GWN	r_s	The private key r_s of the sensor node S_j
SID_j	The identity SID_j of the sensor node S_j	R_i and R_u	The public key R_i and R_u of the user U_i
PW_i	The password PW_i of the user U_i	R_g	The public key R_g of the gateway node GWN
BIO_i	The biological factor BIO_i of the user U_i	R_s	The public key R_s of the sensor node S_j
SC_i	The smart card SC_i of the user U_i	$\cdot$	Elliptic curve point multiplication operation
SK_u	The negotiated session key SK_u of the user U_i	Gen	The generation process of fuzzy extraction
SK_s	The negotiated session key SK_s of the sensor node S_j	Rep	Recovery process of fuzzy extraction
K_G	Gateway node secret value K_G	α_i	The random secret information generated by fuzzy extraction α_i of the user U_i
$List$	Number of user authentication	β_i	The auxiliary bit string generated by fuzzy extraction β_i of the user U_i
$\|\|$	concatenation operator	$h(\cdot)$	hash function
T	Timestamp	$\oplus$	XOR operator
ΔT	Maximum permitted transmission delay	mod	Modular exponentiation

Table 2. Notations used in this paper and descriptions of these notations.

2) Registration. This stage comprises sensor registration and user registration. Both sensor nodes and users are needed to register their fundamental details with the nearest *HGWN* gateway. After the registration, U_i saves $B_1 = h(\alpha_i \| ID_i \| PW_i) \oplus r_i$, $B_2 = h(HPW_i \| \alpha_i \| ID_i \| r_i) mod\ n_0$ to *SC*, *HGWN* saves SID_j, and S_j saves x_j.
3) Login. U_i inputs ID_i, PW_i, and BIO_i, *SC* verifies the identity of U_i by calculating $B_2 = h(HPW_i \| \alpha_i \| ID_i \| r_i) mod\ n_0$, if the verification passes, U_i sends $M_1 = \{TID_i, ID_{hg}, SID_j, D_0, D_1, D_2, D_3, T_1\}$ over the public channel to *HGWN*.
4) Authentication and key agreement.After receiving the communication request between U_i and SID_j, *HGWN* initially verifies if the designated sensor S_j is within its communication range. If *HGWN* can retrieve SID_j from its local database, it can proceed following Case 1, and the three parties, U_i, *HGWN*, and SID_j, perform authentication and key agreement; otherwise, it operates according to Case 2, and the four parties, U_i, *HGWN*, *FGWN*, and SID_j, perform authentication and key agreement.
5) Password update. User enters his or her ID_i, PW_i, and BIO_i, and *SC* verifies. If the verification passes, the user enters new password PW_i', *SC* computes new B_1', B_2', and e_i' and saves.
6) Smart card logout. The user enters his or her ID_i, PW_i, and BIO_i and *SC* verifies it. If the verification passes, U_i sends $M_0 = \{TID_i, \beta_i, R_0, T_1\}$ over the public channel to *HGWN*. *HGWN* verifies that K_i' is equal to K_i by computation. if the verification passes it deletes U_i's information $\{ID_i, K_i, honey\_list\}$.

The existing scheme[16] has some advantages in resisting password guessing, replay, and other attacks to achieve two-way authentication and key agreement; however, there are also security vulnerabilities, such as the inability to guarantee anonymity and the potential to suffer from MITT attacks. In this section, the advantages of the scheme and the existence of security vulnerabilities are presented[21].

Advantages of the scheme[16]

The advantages of the schemes[16] include the following:

1) The use of biometric-based fuzzy extraction technology effectively enhances the security of user login via the three-factor authentication mechanism.
2) Security of the authentication process is ensured through use of the challenge/response mechanism[22].
3) The user's secret x_i and the sensor's secret x_j are calculated using the hash function, and they are not transmitted in the public channel, which can prevent the secret from being cracked and ensure its forward security.
4) The honey list technique, which can prevent password guessing attacks by setting the number of logins and avoid smart card loss attacks and offline guessing attacks, is adopted.
5) Replay attacks are avoided by setting the timestamp *T*.
6) Two-way authentication and key agreement are achieved as the negotiated session key *SK* contains a random number of users, gateways, and sensors to improve the security of the negotiated key[23].

Security vulnerabilities of the scheme[16]

The scheme'[16] security vulnerabilities include the following:

1) Unable to meet the anonymity requirement: During the registration process, U_i sends ID_i to *HGWN*, *Sj* sends SID_j to *HGWN*, and *HGWN* sends ID_{hg} to U_i. Attackers intercept ID_i, ID_{hg}, and SID_j in the public channel to easily obtain the identity ID_s of the user, gateway, and node. Therefore, the scheme cannot guarantee anonymity.
2) Unable to secure user parameters[24]: During the registration process, U_i sends $\{ID_i, HPW_i, \beta_i\}$ to the *HGWN*. The attacker intercepts ID_i in the public channel. During the login process, U_i sends $M_1 = \{TID_i, ID_{hg}, SID_j, D_0, D_1, D_2, D_3, T_1\}$ to the *HGWN*. The attacker intercepts D_2 in the public channel and calculates:

$$h(r_u || x_i) = ID_i \oplus D_2 \tag{1}$$

The attacker intercepts D_0 and calculates:

$$\beta_i = D_0 \oplus h(x_i || r_u) \tag{2}$$

$$K_i = h(ID_i || \beta_i) \tag{3}$$

$$e_i = HPW_i \oplus K_i \oplus x_i \tag{4}$$

The attacker obtains all the parameters of the user login.
3) Unable to secure user secrets x_i and sensor secrets x_j: During the registration process, U_i sends $\{ID_i, HPW_i, \beta_i\}$ to *HGWN* and *HGWN* sends $\{TID_i, \beta_i, e_i, ID_{hg}\}$ to U_i. The attacker intercepts HPW_i, ID_i, β_i, and e_i in the public channel and calculates:

$$K_i = h(ID_i || \beta_i) \tag{5}$$

$$x_i = HPW_i \oplus K_i \oplus e_i \tag{6}$$

The user secret x_i is cracked. Attackers directly obtain sensor secret x_j in the public channel.

4) Unable to secure user private key r_u: During the login process, U_i sends $M_1\{TID_i, ID_{hg}, SID_j, D_0, D_1, D_2, D_3, T_1\}$ to *HGWN*, and the attacker intercepts D_1 in the public channel and can crack x_i by point (3) above and calculates:

$$r_u = D_1 \oplus x_i \tag{7}$$

The user private key r_u is cracked.

5) Unable to secure gateway private key r_{hg} and sensor private key r_s: During the registration process, *HGWN* sends $\{x_j\}$ to S_j. The attacker intercepts x_j in the public channel. During the authentication process, the *HGWN* sends $M_2 = \{D_0, D_4, D_5, D_6, T_2\}$ to S_j and S_j sends $M_3 = \{D_7, D_8, T_3\}$ to the *HGWN*. The attacker intercepts D_4, D_7, T_2, T_4 in the public channel and can crack[25]:

$$r_{hg} = D_4 \oplus h(x_j||T_2) \tag{8}$$

The attacker crack:

$$r_s = D_7 \oplus h(x_j||r_{hg}||T_4) \tag{9}$$

6) Unable to achieve secure two-way authentication: According to Points (2), (3), and (4) above, the attacker cracks x_i, r_u, K_i, During the registration process, U_i sends $\{ID_i, HPW_i, \beta_i\}$ to the *HGWN*, and during the login process, U_i sends $M_1 = \{TID_i, ID_{hg}, SID_j, D_0, D_1, D_2, D_3, T_1\}$ to the *HGWN*. The attacker intercepts TID_i, ID_i, SID_j, T_1 in the public channel, and by calculating $D_3 = h(TID_i\|ID_i\|SID_j\|r_u\|x_i\|K_i\|T_1)$ can crack D_3, so the gateway authentication user algorithm is cracked. During registration, *HGWN* sends $\{x_j\}$ to S_j, during login, U_i sends $M_1 = \{TID_i, ID_{hg}, SID_j, D_0, D_1, D_2, D_3, T_1\}$ to *HGWN*, and during authentication, *HGWN* sends $M_2 = \{D_0, D_4, D_5, D_6, T_2\}$ to S_j. According to Points (4) and (5) above, the attacker cracks r_u, r_{hg} and intercepts SID_j, ID_{hg}, x_j, T_2 in the public channel; D_6 can be cracked by calculating:

$$D_6 = h(SID_j||ID_{hg}||r_u||r_{hg}||x_j||T_2) \tag{10}$$

The sensor authentication gateway algorithm is cracked.

7) Unable to negotiate a secure session key: The negotiated key is $SK_s = h(r_u\|r_{hg}\|r_s\|ID_{hg})$. During the login process, U_i sends $M_1 = \{TID_i, ID_{hg}, SID_j, D_0, D_1, D_2, D_3, T_1\}$ to *HGWN*. According to Points (4) and (5) above, the attacker breaks r_u, r_{hg}, r_s and intercepts ID_{hg} in the public channel, which can crack:

$$SK_s = h(r_u||r_{hg}||r_s||ID_{hg}) \tag{11}$$

The scheme cannot negotiate a secure session key, and it has forward security problems.

8) Unable to resist MITT attacks: The attacker records all $M_1 = \{TID_i, ID_{hg}, SID_j, D_0, D_1, D_2, D_3, T_1\}$ sent to the *GWN*, all $M_2 = \{D_4, D_5, D_6, T_2\}$ sent to S_j, and all x_j sent to S_j by the gateway, and then calculates:

$$r_{hg}^* = D_4 \oplus h(x_j^*||T_2) \tag{12}$$

$$r_u^* = D_5 \oplus h(r_{hg}^*||x_j^*||T_2) \tag{13}$$

For each group M_1, the attacker calculates:

$$x_i^* = r_u^* \oplus D_1 \tag{14}$$

$$\beta_i^* = D_0 \oplus h(x_i^*||r_u^*) \tag{15}$$

$$ID_i^* = D_2 \oplus h(r_u^*||x_i^*) \tag{16}$$

$$K_i^* = h(ID_i^*||\beta_i^*) \tag{17}$$

Whether $D_3^* = h(TID_i\|ID_i^*\|SID_j\|r_u^*\|x_i^*\|K_i^*\|T_1)$ is equal to D_3 is verified. If equal, the attacker can determine user U_i with its corresponding S_j and obtain the values of the parameters r_u, x_i, and so on. The attacker starts a new session with user U_i, selects r_{hg}, r_s, and TID_i', and calculates:

$$SK_{hg} = h(r_u||r_{hg}||r_s||ID_{hg}) \tag{18}$$

$$D_9 = r_s \oplus h(x_i||r_u) \tag{19}$$

$$D_{10} = r_{hg} \oplus h(r_u||x_i) \tag{20}$$

$$x_i' = h(TID_i'||x_{hg}) \oplus R_h \tag{21}$$

$$D_{11} = TID_i' \oplus h(x_i||ID_i||r_u) \tag{22}$$

$$D_{12} = x_i' \oplus h(TID_i'||x_i) \tag{23}$$

$$D_{13} = h(SK_{hg}||x_i'||TID_i'||K_i||T_4) \tag{24}$$

The attacker sends $M_4 = \{D_9, D_{10}, D_{11}, D_{12}, D_{13}, T_4\}$ to U_i. U_i calculates:

$$r_s^* = D_9 \oplus h(x_i||r_u) \tag{25}$$

$$r_{hg}^* = D_{10} \oplus h(r_u||x_i) \tag{26}$$

$$SK_u^* = h(r_u||r_{hg}^*||r_s^*||ID_{hg}) \tag{27}$$

$$TID_i'^* = D_{11} \oplus h(x_i||ID_i||r_u) \tag{28}$$

$$x_i'^* = D_{12} \oplus h(TID_i'^*||x_i) \tag{29}$$

U_i verifies whether $D_{13}{}^* = h(SK_{hg}{}^*\|x'^*\|TID_i'^*\|K_i\|T_4)$ is equal to D_{13}. If equal, according to the rule, the user accepts this SK as the agreement key and the attacker successfully implements the MITT attack.

The proposed scheme

In this section, an ECC-based three-factor authentication and key agreement scheme for WSNs is proposed, the improvement measures of the scheme are introduced, and then a specific implementation scheme, including system initialization, node registration, user registration, two-way authentication and key agreement, password update, and smart card logout, is proposed[17]. The proposed scheme operates under the following security assumptions:

1) The gateway is securely impenetrable and has unlimited computation, storage, and communication capabilities.
2) The WSN network is a bidirectional channel, and nodes can communicate normally.
3) The WSN network employs asymmetric encryption, meaning it utilizes both public and private keys.
4) Upon successful completion of the key agreement in the WSN network, the user and the sensor node can establish communication using the session key.

Scheme improvement measures

1) The authentication scheme is designed using an ECC key agreement protocol to ensure the forward security of the scheme.
2) The user ID is replaced by the user identifier TID after the hashing operation, all IDs are forbidden to be sent explicitly, and no direct XOR calculation can be performed to ensure the anonymity of the scheme.
3) Random numbers r_u and r_s are forbidden to be sent in clear text, and no direct XOR calculation can be performed to ensure secure two-way authentication and key agreement and resist MITT attacks[26].
4) More complex parameters are selected to improve the security of the session key.
5) The relevant parameters in the SC card are updated after two-way authentication and key agreement to ensure that the scheme is resistant to internal attacks[27].

Specific implementation plan

1) System Initialization
   At the very beginning, the system needs to be initialized. GWN selects $E(F_p)$, P, $h(.)$ and the secret value K_G, publicly release $E(F_p)$, P, $h(.)$, save K_G.
2) Node Registration
   After the system is initialized, the node can start registering. Node S_j applies for registration to the GWN, which selects the unique SID_j of the node, calculates $x_j = h(SID_j\|K_G)$, and writes $\{SID_j, x_j\}$ to node S_j.
3) User Registration
   After the system is initialized, the user can start registering. The user registration process is shown in Fig. 1.

   - Step R1: User U_i inputs ID_i, PW_i, BIO_i, chooses random number $r_i \in Z_p^*$, calculates $R_i = r_i \cdot P$, $Gen(BIO_i) = (\alpha_i, \beta_i)$, $TID_i = h(ID_i\|\alpha_i\|r_i)$, $HPW_i = h(PW_i\|\alpha_i)$, and U_i sends $\{TID_i, HPW_i, R_i\}$ to GWN.
   - Step R2: The gateway GWN chooses a random number $r_g \in Z_p^*$ and calculates $R_g = r_g \cdot P$. After the GWN receives the U_i message, it calculates $x_i = h(TID_i\|K_G)$, $K_i = h(TID_i\|HPW_i)$, $R_{ig} = r_g \cdot R_i$, $e_i = x_i \oplus R_{ig} \oplus K_i$, sets the number of logins $List = 0$, saves $\{TID_i, HPW_i, List = 0\}$. Write $\{R_g, e_i\}$ to smart card SC_i and issue to U_i.
   - Step R3: User U_i receives the smart card SC_i, calculates $K_i = h(TID_i\|HPW_i)$, $R_{ig} = r_i \cdot R_g$, $x_i = e_i \oplus R_{ig} \oplus K_i$, $B_1 = h(ID_i\|\alpha_i\|PW_i) \oplus r_i$, $B_2 = h(HPW_i\|ID_i\|\alpha_i\|r_i) mod\ n_0$, and writes $\{B_1, B_2, \beta_i\}$ to the smart card SC_i.

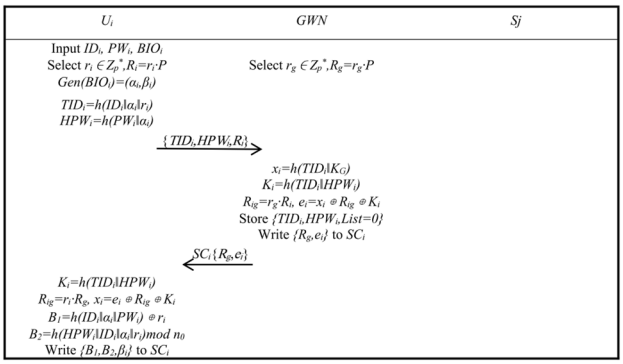

Figure 1. Registration phase.

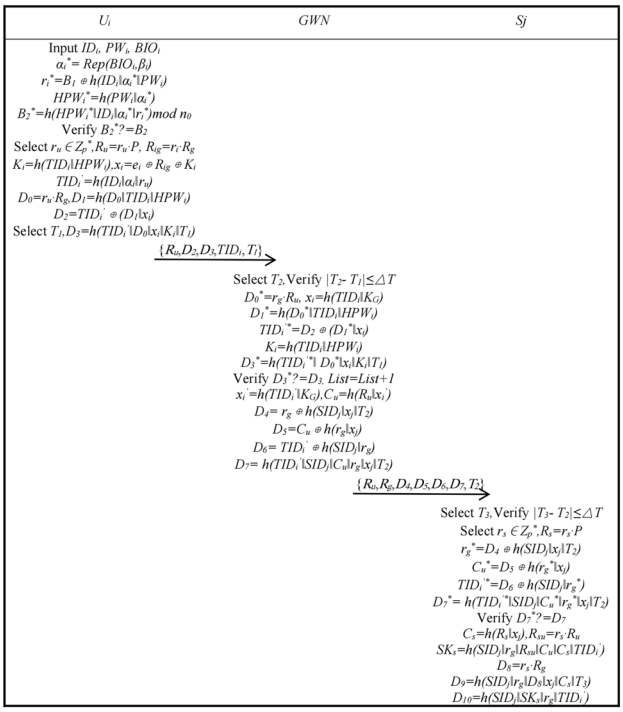

Figure 2. The authentication and key agreement phase 1.

4) Authentication and Key Agreement

After node and user registration is complete, the user, *GWN*, and node can start authentication and key agreement. Figures 2 and 3 shows the authentication and key agreement phase.

- Step A1: User U_i inputs ID_i, PW_i, BIO_i, smart card SC_i calculates $\alpha_i^* = Rep(BIO_i, \beta_i)$, $r_i^* = B_1 \oplus h(ID_i \| \alpha_i^* \| PW_i)$, $HPW_i^* = h(PW_i \| \alpha_i^*)$, $B_2^* = h(HPW_i^* \| ID_i \| \alpha_i^* \| r_i^*) mod\ n_0$, SC_i verifies whether B_2^* is equal to B_2 and continues it is; otherwise, terminate. User U_i chooses a random number $r_u \in Z_p^*$ and calculates $R_u = r_u \cdot P$, $R_{ig} = r_i \cdot R_g$, $K_i = h(TID_i \| HPW_i)$, $x_i = e_i \oplus R_{ig} \oplus K_i$, $TID_i^{'} = h(ID_i \| \alpha_i \| r_u)$, $C_u = h(R_u \| x_i^{'})$, $D_0 = r_u \cdot R_g$, $D_1 = h(D_0 \| TID_i \| HPW_i)$, $D_2 = TID_i^{'} \oplus (D_1 \| x_i)$, choose time T_1, calculate $D_3 = h(TID_i^{'} \| D_0 \| x_i \| K_i \| T_1)$. U_i sends $\{R_u, D_2, D_3, TID_i, T_1\}$ to the *GWN*.
- Step A2: The gateway *GWN* receives the message and selects T_2, verifies whether $|T_2 - T_1|$ is less than or equal to ΔT and continues if it is, otherwise terminates. The *GWN* calculates $D_0^* = r_g \cdot R_u$, $x_i = h(TID_i \| K_G)$, $D_1^* = h(D_0^* \| TID_i \| HPW_i)$, $TID_i^{'*} = D_2 \oplus (D_1^* \| x_i)$, $K_i = h(TID_i \| HPW_i)$, $D_3^* = h(TID_i^{'*} \| D_0^* \| x_i \| K_i \| T_1)$, verifies whether D_3^* is equal to D_3 and continues if it is, *List* plus one; otherwise, it is terminated. *GWN* calculates $x_i^{'*} = h(TID_i^{'*} \| K_G)$, $C_u^* = h(R_u \| x_i^{'*})$, $D_4 = r_g \oplus h(SID_j \| x_j \| T_2)$, $D_5 = C_u \oplus h(r_g \| x_j)$, $D_6 = TID_i^{'} \oplus h(SID_j \| r_g)$, $D_7 = h(TID_i^{'} \| SID_j \| C_u \| r_g \| x_j \| T_2)$, and the *GWN* sends $\{R_u, R_g, D_4, D_5, D_6, D_7, T_2\}$ to S_j.
- Step A3: The sensor S_j receives the message and selects T_3, verifies whether $|T_3 - T_2|$ is less than or equal to ΔT and continues it is; otherwise, it is terminated. S_j selects a random number $r_s \in Z_p^*$, calculates $R_s = r_s \cdot P$, $r_g^* = D_4 \oplus h(SID_j \| x_j \| T_2)$, $C_u^* = D_5 \oplus h(r_g^* \| x_j)$, $TID_i^{'*} = D_6 \oplus h(SID_j \| r_g^*)$, $D_7^* = h(TID_i^{'*} \| SI$

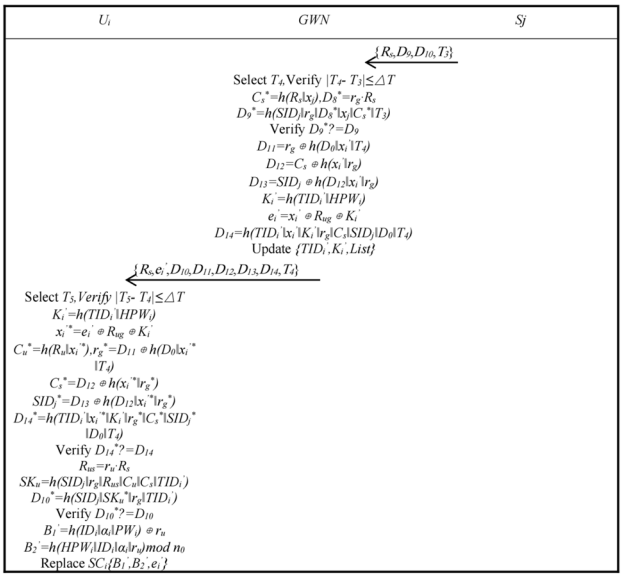

Figure 3. The authentication and key agreement phase 2.

$D_j\|C_u^*\|r_g^*\|x_j\|T_2$), verifies whether D_7^* is equal to D_7 and continues if it is; otherwise, it is terminated. $C_s=h(R_s\|x_j)$, $R_{su}=r_s \cdot R_u$, $SK_s=h(SID_j\|r_g\|R_{su}\|C_u\|C_s\|TID_i')$, $D_8=r_s \cdot R_g$, $D_9=h(SID_j\|r_g\|D_8\|x_j\|C_s\|T_3)$, $D_{10}=h(SID_j\|SK_s\|r_g\|TID_i')$ is calculated, and S_j sends $\{R_s, D_9, D_{10}, T_3\}$ to the *GWN*.

- Step A4: The gateway *GWN* receives the message and selects T_4, verifies whether $|T_4 - T_3|$ is less than or equal to ΔT and continues if it is; otherwise, it is terminated. The *GWN* calculates $C_s^*=h(R_s\|x_j)$, $D_8^*=r_g \cdot R_s$, $D_9^*=h(SID_j\|r_g\|D_8^*\|x_j\|C_s^*\|T_3)$, verifies whether D_9^* is equal to D_9 and continues if it is; otherwise, it is terminated. $D_{11}=r_g \oplus h(D_0\|x_i'\|T_4)$, $D_{12}=C_s \oplus h(x_i'\|r_g)$, $D_{13}=SID_j \oplus h(D_{12}\|x_i'\|r_g)$, $K_i'=h(TID_i'\|HPW_i)$, $e_i'=x_i' \oplus R_{ug} \oplus K_i'$, $D_{14}=h(TID_i'\|x_i'\|K_i'\|r_g\|C_s\|SID_j\|D_0\|T_4)$ is calculated and $\{TID_i', K_i', List\}$ is updated, and the *GWN* sends $\{R_s, e_i', D_{10}, D_{11}, D_{12}, D_{13}, D_{14}, T_4\}$ to U_i.
- Step A5: User U_i receives the message and selects T_5, verifies whether $|T_5 - T_4|$ is less than or equal to ΔT and continues it is; otherwise, it is terminated. U_i calculates $K_i'=h(TID_i'\|HPW_i)$, $x_i'^*=e_i' \oplus R_{ug} \oplus K_i'$, $C_u^*=h(R_u\|x_i'^*)$, $r_g^*=D_{11} \oplus h(D_0\|x_i'^*\|T_4)$, $C_s^*=D_{12} \oplus h(x_i'^*\|r_g^*)$, $SID_j^*=D_{13} \oplus h(D_{12}\|x_i'^*\|r_g^*)$, $D_{14}^*=h(TID_i'\|x_i'^*\|K_i'\|r_g^*\|C_s^*\|SID_j^*\|D_0\|T_4)$, verifies whether D_{14}^* is equal to D_{14} and continues if equal; otherwise, it is terminated. $R_{us}=r_u \cdot R_s$, $SK_u=h(SID_j\|r_g\|R_{us}\|C_u\|C_s\|TID_i')$, $D_{10}^*=h(SID_j\|SK_u\|r_g\|TID_i')$ is calculated, whether D_{10}^* is equal to D_{10} is verified, and it continues if it is; otherwise, it is terminated. This completes the two-way authentication and negotiates the session key *SK* for user U_i and sensor S_j. Finally, U_i calculates $B_1'=h(ID_i\|\alpha_i\|PW_i) \oplus r_u$, $B_2'=h(HPW_i\|ID_i\|\alpha_i\|r_u) mod\ n_0$ with B_1', B_2', e_i' replacing B_1, B_2, e_i within the smart card SC_i.

5) Password Update.

Users can also perform a password update at any time after completing the authentication and key agreement. The password update process is shown in Fig. 4.

- Step P1: User U_i inputs ID_i, PW_i, BIO_i, smart card SC_i calculates $\alpha_i^*=Rep(BIO_i,\beta_i)$, $r_u^*=B_1 \oplus h(ID_i\|\alpha_i^*\|PW_i)$, $HPW_i^*=h(PW_i\|\alpha_i^*)$, $B_2^*=h(HPW_i^*\|ID_i^*\|\alpha_i^*\|r_u^*) mod\ n_0$, verifies whether B_2^* is equal to B_2 and continues if it is; otherwise, it is terminated. SC_i calculates $TID_i=h(ID_i\|\alpha_i\|r_u)$, $K_i=h(TID_i\|HPW_i)$, $R_{ug}=r_u \cdot R_g$, $x_i=e_i \oplus R_{ug} \oplus K_i$.
- Step P2: User U_i enters the new password PW_i^{new}, smart card SC_i calculates $HPW_i^{new}=h(PW_i^{new}\|\alpha_i)$, $K_i^{new}=h(TID_i\|HPW_i^{new})$, $e_i^{new}=R_{ug} \oplus K_i^{new} \oplus x_i$, $B_1^{new}=h(ID_i\|\alpha_i\|PW_i^{new}) \oplus r_u$, $B_2^{new}=h(HPW_i^{new}\|ID_i\|\alpha_i\|r_u) mod\ n_0$, replacing B_1,B_2,e_i in smart card SC_i with B_1^{new}, B_2^{new}, e_i^{new}, and the password update is completed.

6) Smart Card Logout

Smart Card Logout can be performed when the user's Smart Card is no longer in use. The smart card logout process is shown in Fig. 5.

- Step S1: User U_i inputs ID_i, PW_i, BIO_i, calculates $\alpha_i^*=Rep(BIO_i,\beta_i)$, $r_u^*=B_1 \oplus h(ID_i\|\alpha_i^*\|PW_i)$, $HPW_i^*=h(PW_i\|\alpha_i^*)$, $B_2^*=h(HPW_i^*\|ID_i\|\alpha_i^*\|r_u^*) mod\ n_0$, verifies whether B_2^* is equal to B_2 and continues if it is; otherwise, it is terminated. $K_i=h(TID_i\|HPW_i)$, $R_{ug}=r_u \cdot R_g$, $x_i=e_i \oplus R_{ug} \oplus K_i$ is calculated, time T_1 is chosen, $L_o=x_i \oplus h(K_i\|T_1)$ is calculated, and U_i sends $\{TID_i, L_o, T_1\}$ to the *GWN*.
- Step S2: The gateway *GWN* receives the message and selects T_2, verifies whether $|T_2 - T_1|$ is less than or equal to ΔT and continues if it is; otherwise, it is terminated. The *GWN* calculates $K_i'=h(TID_i\|HPW_i)$, $x_i^*=L_o \oplus h(K_i'\|T_1)$, $x_i=h(TID_i\|K_G)$, verifies whether x_i^* is equal to x_i and continues if it is; otherwise,

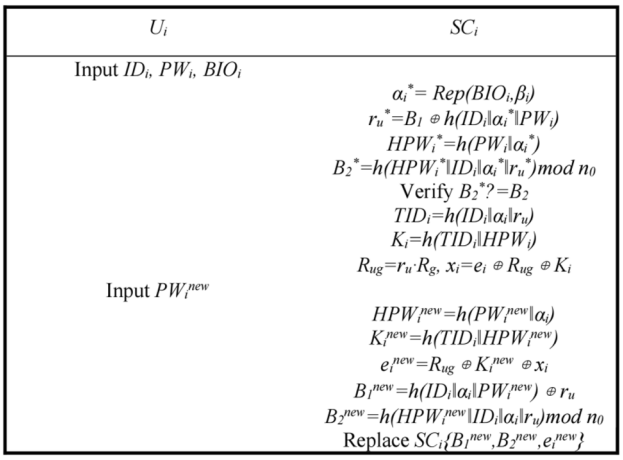

Figure 4. Password update.

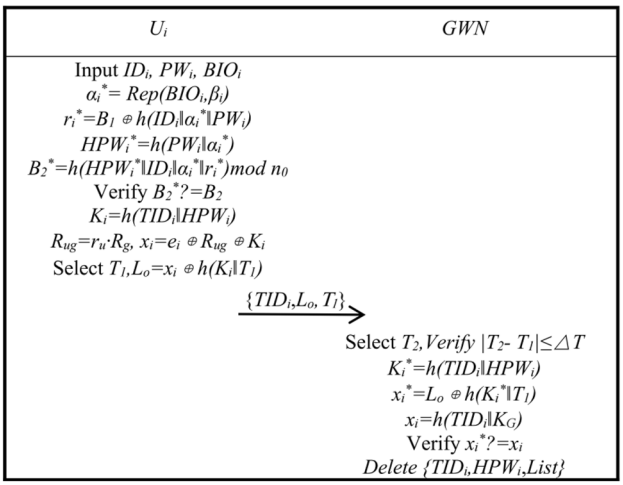

Figure 5. Smart card logout.

it is terminated. Finally, the messages associated with $U_i\{TID_i, HPW_i, List\}$ are deleted, and smart card revocation is completed.

Security analysis

This section provides a formal security analysis of the scheme using BAN logic. The informal security analysis is performed through Propositions 1 to 11 for a variety of known attacks. The security analysis proves the correctness of the scheme; it can resist various security attacks and has high security characteristics[28].

Formal analysis based on BAN logic

Next, BAN logic is used to demonstrate the security of the scheme.

1) Goals
   $G1: S_j| \equiv U_i \overset{SK}{\leftrightarrow} S_j$ $G2: S_j| \equiv U_i| \equiv U_i \overset{SK}{\leftrightarrow} S_j$
   $G3: U_i| \equiv S_j \overset{SK}{\leftrightarrow} U_i$ $G4: U_i| \equiv S_j| \equiv S_j \overset{SK}{\leftrightarrow} U_i$
2) Idealized Forms
   $M1: U_i \rightarrow GWN : R_u, D_2, T_1, TID_i, < TID_i', D_0, k_i >_{x_i}$
   $M2: GWN \rightarrow S_j : R_u, R_g, D_4, D_5, D_6, T_2, < TID_i', U_i| \equiv C_u, r_g >_{x_j}$
   $M3: S_j \rightarrow GWN : R_s, D_{10}, T_3, < D_8, r_g, S_j| \equiv C_s >_{x_j}$
   $M4: GWN \rightarrow U_i : e_i', R_s, D_{10}, D_{11}, D_{12}, D_{13}, T_4, < TID_i', x_i', D_0, r_g, S_j| \equiv C_s >_{k_i'}$
3) Assumptions
   $A1: GWN| \equiv U_i \overset{x_i}{\leftrightarrow} GWN$ $A2: S_j| \equiv GWN \overset{x_j}{\leftrightarrow} S_j$

$A3: GWN| \equiv S_j \overset{x_j}{\leftrightarrow} GWN$ $A4: U_i| \equiv GWN \overset{k_i'}{\leftrightarrow} U_i$
$A5: GWN| \equiv \#(C_u)$ $A6: S_j| \equiv \#\left(r_g\right)$
$A7: GWN| \equiv \#(C_s)$ $A8: U_i| \equiv \#\left(r_g\right)$
$A9: GWN| \equiv U_i| \Rightarrow < D_3 >$ $A10: S_j| \equiv GWN| \Rightarrow < D_7 >$
$A11: GWN\left| \equiv S_j\right| \Rightarrow < D_9 >$ $A12: U_i| \equiv GWN| \Rightarrow < D_{14} >$
$A13: S_j| \equiv \#(C_u)$ $A14: U_i| \equiv \#(C_s)$
$A15: S_j| \equiv U_i| \sim U_i \overset{SK}{\leftrightarrow} S_j$ $A16: U_i\left| \equiv S_j\right| \sim U_i \overset{SK}{\leftrightarrow} S_j$

4) Main Proofs

From *M1*, they can get *S1*: $GWN \lhd < D_3 >_{x_i}$.
From *S1*, *A1*, *R1*, they can get *S2*: $GWN| \equiv U_i| \sim < D_3 >$.
From *A5*, *R4*, they can get *S3*: $GWN| \equiv \#(< D_3 >)$.
From *S2*, *S3*, *R2*, they can get *S4*: $GWN| \equiv U_i| \equiv < D_3 >$.
From *S4*, *A9*, *R3*, they can get *S5*: $GWN| \equiv < D_3 >$.
From *M2*, they can get *S6*: $S_j \lhd < D_7 >_{x_j}$.
From *S6*, *A2*, *R1*, they can get *S7*: $S_j| \equiv GWN| \sim < D_7 >$.
From *A6*, *R4*, they can get *S8*: $S_j| \equiv \#(< D_7 >)$.
From *S7*, *S8*, *R2*, they can get *S9*: $S_j| \equiv GWN| \equiv < D_7 >$.
From *S9*, *A10*, *R3*, they can get *S10*: $S_j| \equiv < D_7 >$.
From *S10*, *R5*, they can get *S11*: $S_j| \equiv U_i| \equiv C_u$.

$$SK = h\left(SID_j||r_g||R_{su}||C_u||C_s||TID_i'\right).$$

From *S11*, *A13*, *SK*, *R6*, they can get *S12*: $S_j| \equiv U_i \overset{SK}{\leftrightarrow} S_j$, they have achieved *G*1.
From *S12*, *A13*, *A15*, *R2*, *R4*, they can get S13: $S_j| \equiv U_i| \equiv U_i \overset{SK}{\leftrightarrow} S_j$, they have achieved *G*2.
From *M3*, they can get *S14*: $GWN \lhd < D_9 >_{x_j}$.
From *S14*, *A3*, *R1*, they can get *S15*: $GWN\left| \equiv S_j\right| \sim < D_9 >$.
From *A7*, *R4*, they can get *S16*: $GWN| \equiv \#(< D_9 >)$.
From *S15*, *S16*, *R2*, they can get *S17*: $GWN\left| \equiv S_j\right| \equiv < D_9 >$.
From *S17*, *A11*, *R3*, they can get *S18*: $GWN| \equiv < D_9 >$.
From *M4*, they can get *S19*: $U_i \lhd < D_{14} >_{k_i'}$.
From *S19*, *A4*, *R1*, they can get *S20*: $U_i| \equiv GWN| \sim < D_{14} >$.
From *A8*, *R4*, they can get *S21*: $U_i| \equiv \#(< D_{14} >)$.
From *S20*, *S21*, *R2*, they can get *S22*: $U_i| \equiv GWN| \equiv < D_{14} >$.
From *S22*, *A12*, *R3*, they can get *S23*: $U_i| \equiv < D_{14} >$.
From *S23*, *R5*, they can get *S24*: $U_i\left| \equiv S_j\right| \equiv C_s$.

$$SK = h\left(SID_j||r_g||R_{us}||C_u||C_s||TID_i'\right).$$

From *S24*, *A14*, *SK*, *R6*, they can get *S25*: $U_i| \equiv S_j \overset{SK}{\leftrightarrow} U_i$, they have achieved *G*3.
From *S25*, *A14*, *A16*, *R2*, *R4*, they can get S26: $U_i| \equiv S_j| \equiv S_j \overset{SK}{\leftrightarrow} U_i$, they have achieved *G*4.

In summary, according to the BAN logic rules, the security objectives *G1* to *G4* of this scheme have been achieved, and the security of the scheme has been proven.

Formal analysis based on the random oracle model

Theorem 1 In a scenario where an adversary attacker (*A*) operates within probabilistic polynomial time (PPT) against a protocol (*P*) in a random oracle, *A* is allowed to make up to q_s Send ($\prod^{*}_{I}, m$) queries, q_e Execute ($\prod^{i}_{U}, \prod^{k}_{GWN}, \prod^{j}_{S}$) queries, and q_h oracle queries. Let *D* denote the password space, which follows a Zipf distribution with parameters C' and s'[16]. Additionally, *l* represents the output length of the hash function and AKE represents authenticated key agreement. In the context of the random oracle model, the probability *P* of *A* successfully compromising the protocol in PPT is defined as follows:

$$\mathrm{Adv}_{\mathrm{p}}^{\mathrm{AKE}}(\mathrm{A}) = 2|\Pr[\mathrm{S}_4] - \Pr[\mathrm{S}_0]| \le \max\left\{\frac{\mathrm{q_s}}{2^{\mathrm{l}_\alpha - 1}}, 2\mathrm{C}'\mathrm{q_s}^{\mathrm{s}'}, \frac{\mathrm{q_s}}{2^{\mathrm{l}-1}}\right\} + \frac{\mathrm{q_s}}{2^{\mathrm{l}-1}} + \frac{\mathrm{q_h^2}}{2^{\mathrm{l}}} + \frac{\left(\mathrm{q_s} + \mathrm{q_e}\right)^2}{\mathrm{p} - 1} \tag{30}$$

Proof: The scheme is divided into five games, labelled $G_i (i = 1, 2, 3, 4, 5)$. In each game, there is a condition denoted as S_i, indicating that *A* successfully predicts a bit *b* before advancing in the game.

G_0: It mimics a real attack in the random oracle model, where *A* has full access to all oracles. Hence,

$$\mathrm{Adv}_{\mathrm{p}}^{\mathrm{AKE}}(\mathrm{A}) = 2\Pr[\mathrm{S}_0] - 1 \tag{31}$$

```
(*--------Channels-------*)
free sch1:channel .    (*the channel between Ui and GWN*)
free sch2:channel .    (*the channel between Sj and GWN*)
free ch1:channel .     (*the channel between Ui and GWN*)
free ch2:channel .     (*the channel between Sj and GWN*)
(*--------variables and constants-------*)
free SKu:bitstring [private].
free SKs:bitstring [private].
const P:bitstring.
const Bio:bitstring.
const Bio':bitstring.
const n0:bitstring.
const KG: bitstring [private].
const xi: bitstring [private].
const xj: bitstring [private].
free IDi: bitstring [private].
free SIDj: bitstring [private].
(*--------operations-------*)
fun con (bitstring, bitstring):bitstring.
fun con3 (bitstring, bitstring, bitstring):bitstring.
fun con4 (bitstring, bitstring, bitstring, bitstring):bitstring.
fun con5 (bitstring, bitstring, bitstring, bitstring, bitstring):bitstring.
fun con6 (bitstring, bitstring, bitstring, bitstring, bitstring, bitstring):bitstring.
fun con7 (bitstring, bitstring, bitstring, bitstring, bitstring, bitstring, bitstring):bitstring.
fun con8 (bitstring, bitstring, bitstring, bitstring, bitstring, bitstring, bitstring, bitstring):bitstring.
fun mult (bitstring, bitstring):bitstring.
fun syme (bitstring, bitstring):bitstring.
reduc forall m:bitstring, key:bitstring; symd (syme (m,key), key) = m.
fun mod (bitstring, bitstring):bitstring.
fun xor (bitstring, bitstring):bitstring.
fun xor3 (bitstring, bitstring, bitstring):bitstring.
fun h (bitstring):bitstring.
fun Gen (bitstring):bitstring.
fun Rep (bitstring, bitstring):bitstring.
(*--------events-------*)
event beginSj (bitstring).
event endSj (bitstring).
event beginUi (bitstring).
event endUi (bitstring).
```

Figure 6. Define the channels, variables, constants, operations and events.

G_1: In G_1, A conducts a passive attack, intercepting messages through the Excute(*) query and attempting to guess the output of the Test ($\prod_S^j$) query. However, the impossibility of deducing $SK = h(SID_j \| r_g \| R_{us} \| C_u \| C_s \| TID_i')$ means that A's advantage in a successful attack does not increase. Hence,

$$\Pr[S_1] = \Pr[S_0] \tag{32}$$

G_2: A is allowed to make Send ($\prod_I^*, m$) and H queries to persuade the legitimate communicator with forged messages. The simulation concludes only if A manages to discover collisions and successfully constructs convincing messages. The probabilities of their occurrence, based on the birthday paradox[29], are ($q_h^2/2^{l+1}$) and (($q_s + q_e$)^2/2(p-1)). Hence,

$$|\Pr[S_2] - \Pr[S_1]| \leq \frac{q_h^2}{2^{l+1}} + \frac{(q_s + q_e)^2}{2(p-1)} \tag{33}$$

G_3: This game is distinct from the earlier games because if A successfully guesses the correct authentication Factors D_3, D_7, D_9, and D_{14}. The simulation concludes if H queries are not utilized. It is identical to the previous games in all aspects, except for situations where correct authentication is refused. Hence,

$$|\Pr[S_3] - \Pr[S_2]| \leq \frac{q_s}{2^l} \tag{34}$$

G_4: In this game, A can acquire more information through the Corrupt ($\prod_U^i, a$) query. A successfully guesses α_i with a length of l_α, with a probability of ($q_s/2^l{}_\alpha$). Additionally, A successfully guesses the victim's password with a probability of $C' q_s^{s'}$. The likelihood of A guessing the correct x_i is ($q_s/2^l$). Hence,

$$|\Pr[S_4] - \Pr[S_3]| \leq \max\left\{\frac{q_s}{2^{l_\alpha}}, C' q_s^{s'}, \frac{q_s}{2^l}\right\} \tag{35}$$

$$\Pr[S_4] = \frac{1}{2} \tag{36}$$

Based on Eqs. (31) to (36), they can infer either Conclusion (30) or Conclusion (37):

$$\mathrm{Adv}_P^{AKE}(A) = 2|\Pr[S_4] - \Pr[S_0]| \leq \max\left\{\frac{q_s}{2^{l_\alpha - 1}}, 2C' q_s^{s'}, \frac{q_s}{2^{l-1}}\right\} + \frac{q_s}{2^{l-1}} + \frac{q_h^2}{2^l} + \frac{(q_s + q_e)^2}{p-1} \tag{37}$$

Formal security verification via ProVerif[30]

This section presents the formal security verification of the proposed scheme by using the Pi calculus-based simulation tool ProVerif. To date, ProVerif has been used to verify many protocols and demonstrate their correctness

```
(*--------the process of Ui-------*)
let processUi = new PWi:bitstring;
let (alpha_i:bitstring, beta_i:bitstring) = Gen
(Bio) in
new ri: bitstring;
let Ri = mult (ri, P) in
let TIDi = h (con3 (IDi, alpha_i, ri)) in
let HPWi = h (con (PWi, alpha_i)) in
out (sch1, (TIDi, HPWi, Ri));
in (sch1, (Rg:bitstring, ei:bitstring));
let Ki = h (con (TIDi, HPWi)) in
let Rig = mult (ri, Rg) in
let xi = xor3 (ei, Rig, Ki) in
let B1 = xor (h (con3 (IDi, alpha_i, PWi)), ri) in
let B2 = h (mod (h (con4 (HPWi, IDi, alpha_i,
ri)),n0)) in
! (
event beginUi (IDi);
let alpha_i' = Rep (Bio',beta_i) in
let ri' = xor (B1, h (con3 (IDi, alpha_i, PWi)))
in
let HPWi' = h (con(PWi, alpha_i')) in
let B2' = h (mod (h (con4 (HPWi', IDi, alpha_i',
ri')),n0)) in
if B2' = B2 then
new ru:bitstring;
let Ru = mult (ru, P) in
let Rig = mult (ri, Rg) in
let Rug = mult (ru, Rg) in
new T1:bitstring;
let Ki = h (con(TIDi, HPWi)) in
let xi = xor3 (ei, Rig, Ki) in
let TIDii = h (con3 (IDi, alpha_i, ru)) in
let D0 = mult (ru, Rg) in
let D1 = h (con3 (D0, TIDi, HPWi)) in
let D2 = xor (TIDii, h (con(D1, xi))) in
let D3 = h (con5 (TIDii, D0, xi, Ki, T1)) in
out (ch1, (Ru, D2, D3, TIDi, T1));
in (ch1, (Rs:bitstring, eii:bitstring,
D10:bitstring, D11:bitstring, D12:bitstring,
D13:bitstring, D14:bitstring, T4:bitstring));
new T5:bitstring;
let Kii = h (con(TIDii, HPWi)) in
let xii' = xor3 (eii, Rug, Kii) in
let Cu' = h (con(Ru, xii')) in
let rg' = xor (D11, h (con3(D0, xii', T4))) in
let Cs' = xor (D12, h (con(xii', rg'))) in
let SIDj' = xor (D13, h (con3(D12, xii', rg'))) in
let D14' = h (con8 (TIDii, xii', Kii, rg', Cs',
SIDj', D0, T4)) in
if D14' = D14 then
let xii = xor3 (eii, Rug, Kii) in
let Cu = h (con(Ru, xii)) in
let rg = xor (D11, h (con3(D0, xii, T4))) in
let Cs = xor (D12, h (con(xii, rg))) in
let Rus = h (con(ru, Rs)) in
let SKu = h (con6 (SIDj, rg, Rus, Cu, Cs,
TIDii)) in
let D10' = h (con4 (SIDj, SKu, rg, TIDii)) in
if D10' = D10 then
event endSj (SIDj);
let B1_new = xor (h (con3 (IDi, alpha_i, PWi)),
ru) in
let B2_new = h (mod (h (con4 (HPWi, IDi,
alpha_i, ru)),n0)) in
0
).
```

Figure 7. The process of U_i.

```
(*--------the process of GWN-------*)
let processGWN = in (sch1, (TIDi:bitstring,
HPWi:bitstring, Ri:bitstring));
new rg: bitstring;
let Rg = mult (rg, P) in
let xi = h (con (TIDi, KG)) in
let Ki = h (con (TIDi, HPWi)) in
let Rig = mult (rg, Ri) in
let ei = xor3 (xi, Rig, Ki) in
out (sch2, (SIDj, xj));
out (ch1, (Rg, ei));
! (
in (ch1, (Ru:bitstring, D2:bitstring,
D3:bitstring, TIDi:bitstring, T1:bitstring));
let D0' = mult (rg, Ru) in
let xi = h (con (TIDi, KG)) in
let D1' = h (con3 (D0', TIDi, HPWi)) in
let TIDii' = xor (D2, h (con(D1', xi))) in
let Ki = mult (TIDi, HPWi) in
let D3' = h (con5 (TIDii',D0',xi, Ki,T1)) in
if D3' = D3 then
new T2:bitstring;
let D0 = mult (rg, Ru) in
let D1 = h (con3 (D0, TIDi, HPWi)) in
let TIDii = xor (D2, h (con(D1, xi))) in
let xii = h (con (TIDii, KG)) in
let Cu = h (con (Ru, xii)) in
let D4 = xor (rg, h (con3(SIDj, xj, T2))) in
let D5 = xor (Cu, h (con(rg, xj))) in
let D6 = xor (TIDii, h (con(SIDj, rg))) in
let D7 = h (con6 (TIDii',SIDj,Cu, rg, xj, T2)) in
out (ch2, (Ru, Rg, D4, D5, D6, D7, T2));
in (ch2, (Rs:bitstring, D9:bitstring,
D10:bitstring, T3:bitstring));
let Cs' = h (con (Rs, xj)) in
let D8' = mult (rg, Rs) in
let D9' = h (con6 (SIDj, rg, D8', xj, Cs', T3)) in
if D9' = D9 then
new T4:bitstring;
let D11 = xor (rg, h (con3 (D0, xii, T4))) in
let Cs = h (con (Rs, xj)) in
let D12 = xor (Cs, h (con (xii, rg))) in
let D13 = xor (SIDj, h (con3 (D12, xii, rg ))) in
let Kii = mult (TIDii, HPWi) in
new ru:bitstring;
let Rug = mult (ru, Rg) in
let eii = xor3 (xi, Rug, Kii) in
let D14 = h (con8 (TIDii, xii, Kii, rg, Cs, SIDj,
D0, T4)) in
out (ch1, (Rs, eii, D10, D11, D12, D13, D14,
T4));
0
).
```

Figure 8. The process of *GWN*.

```
(*--------the process of Sj-------*)
let processSensor = in (sch2, (SIDj:bitstring,
xj:bitstring));
! (
event beginSj (SIDj);
in (ch2, (Ru:bitstring, Rg:bitstring,
D4:bitstring, D5:bitstring, D6:bitstring,
D7:bitstring, T2:bitstring));
new rs:bitstring;
let xj = h (con(SIDj, KG)) in
let Rs = mult (rs, P) in
let rg' = xor (D4, h (con3 (SIDj, xj, T2))) in
let Cu' = xor (D5, h (con (rg', xj))) in
let TIDii' = xor (D6, h (con (SIDj, rg'))) in
let D7' = h (con6 (TIDii', SIDj, Cu', rg', xj, T2))
in
if D7' = D7 then
event endUi (IDi);
new T3:bitstring;
let Cs = h (con(Rs, xj)) in
let Rsu = mult (rs, Ru) in
let rg = xor (D4, h (con3 (SIDj, xj, T2))) in
let TIDii =    xor (D6, h (con (SIDj, rg))) in
let Cu = xor (D5, h (con (rg, xj))) in
let SKs = h (con6 (SIDj, rg, Rsu, Cu, Cs,
TIDii)) in
let D8 = mult (rs, Rg) in
let D9 = h (con6 (SIDj, rg, D8, xj, Cs, T3)) in
let D10 = h (con4 (SIDj, SKs, rg, TIDii)) in
out (ch2, (Rs, D9, D10, T3));
0
).
```

Figure 9. The process of S_j.

```
(*--------query-------*)
query attacker (SKu).
query attacker (SKs).
query id:bitstring; inj-event (endSj (id)) ==>inj-event(beginSj (id)).
query id:bitstring; inj-event (endUi (id)) ==>inj-event(beginUi (id)).
process !processUi | !processGWN | !processSensor
```

Figure 10. Define the queries and simulate the scheme.

```
--------------------------------------------------------------
Verification summary:

Query not attacker(SKu[]) is true.

Query not attacker(SKs[]) is true.

Query inj-event(endSj(id)) ==> inj-event(beginSj(id)) is true.

Query inj-event(endUi(id)) ==> inj-event(beginUi(id)) is true.

--------------------------------------------------------------
```

Figure 11. Outputs of the Proverif verification.

and robust properties, so ProVerif is used in this study to rectify the secrecy and authentication properties of the focal protocol.

The channels, variables, constants, operations and events are defined as shown in Fig. 6:

According to the proposed scheme execution, they define the process of U_i as shown in Fig. 7:

The process of *GWN* is modeled as shown in Fig. 8:

The process of S_j is modeled as shown in Fig. 9:

The queries are defined and the whole scheme is simulated as executing in parallel as shown in Fig. 10:

The outputs of the ProVerif verification is shown in Fig. 11:

Results (1) and (2) indicate the secrecy of the proposed scheme because of the failing query attack on session keys SK_S and SK_U. Moreover, Results (3) and (4) confirm the successful mutual authentication between U_i and S_j. In other words, the proposed scheme not only provides the secrecy of the session key, but also achieves the authentication property by verifying the correspondence assertions in the Dolev-Yao model.

Informal analysis

This scheme can resist many common attacks and effectively address the shortcomings of existing schemes. The proof of this is as follows:

Proposition 1 *The scheme has anonymity.*

Proof All identity ID in the scheme are not transmitted in clear text in the public channel, and the identity identifiers $TID_i = h(ID_i \| \alpha_i \| r_i)$ and $TID_i^{'} = h(ID_i \| \alpha_i \| r_u)$ are used to replace the ID for transmission[17]. Assuming that the attacker intercepts TID_i, according to the one-way property of the hash function, the attacker cannot resolve ID_i[31]. In addition, even if the attacker intercepts both TID_i and $TID_i^{'}$, it is impossible to determine whether the two parameters come from the same ID; hence, the scheme has anonymity.

Proposition 2 *The scheme is resistant to registered legitimate user attacks.*

Proof Suppose attacker U_a registers legitimate user ID_a and calculates $TID_a = h(ID_a \| \alpha_a \| r_a)$. U_a registers with gateway GWN, which calculates $x_a = h(TID_a \| K_G)$, $K_a = h(TID_a \| HPW_a)$. The TID_a generated by the attacker based on ID_a is different from the TID_s of other legitimate users, and the x and K generated by registering to GWN through TID_a are also different. Therefore, the scheme can resist registered legitimate user attacks by generating new identity information TID_s, and the attacker cannot obtain messages to any other legitimate user by registering a legitimate user.

Proposition 3 *The scheme is resistant to smart card loss attacks and offline guessing attacks*[17].

Proof Suppose that a user's smart card is lost or stolen, and the attacker obtains the card and the information it contains, $B_1 = h(ID_i \| \alpha_i \| PW_i) \oplus r_i$, $B_2 = h(HPW_i \| ID_i \| \alpha_i \| r_i) mod\ n_0$, by differential energy attack, because B_1 and B_2 are hash functions with one-way security. However, the attacker is unable to extract the password PW_i of user U_i from it. Second, if the attacker wishes to obtain the user's password PW_i through offline password guessing, he or she needs to have the biometric trait α_i and the private key r_i, however, the attacker is not in possession of α_i and r_i, and therefore, the attacker is unable to carry out an offline password guessing attack[32]. Again, $B_2 = h(HPW_i \| ID_i \| \alpha_i \| r_i) mod\ n_0$, when n_0 is taken large enough, the number of password guesses grows exponentially and it is not feasible to obtain the password by offline guessing. Finally, the gateway records the number of user authentication $List$, and it is impossible for an attacker to complete an offline guessing attack within a limited number of guesses. Therefore, the scheme resists smart card loss attacks and offline guessing attacks by means of hash functions, biometrics, modulo arithmetic, and recording the number of authentication times, which are infeasible regardless of whether the attacker tries to extract the password from the smart card or crack the password through offline guessing.

Proposition 4 *The scheme is resistant to spoofed user attacks.*

Proof To disguise a user login gateway, the attacker needs to send $\{R_u, D_2, D_3, TID_i, T_1\}$ to the gateway, where $R_u = r_u \cdot P$, $TID_i^{'} = h(ID_i \| \alpha_i \| r_u)$, $C_u = h(R_u \| x_i^{'})$, $D_0 = r_u \cdot R_g$, $D_1 = h(D_0 \| TID_i \| HPW_i)$, $D_2 = TID_i^{'} \oplus (D_1 \| x_i)$, $D_3 = h(TID_i^{'} \| D_0 \| C_u \| x_i \| K_i \| T_1)$; the attacker needs to master the user's private key r_u, identifier TID_i, password PW_i, biometric α_i, secret x_i, key parameters K_i, and so on, so it is clear that the attacker cannot master the above parameters at the same time and cannot make a spoofed user attack. Therefore, the scheme can resist spoofed user attacks by setting various parameters.

Proposition 5 *The scheme is resistant to internal attacks.*

Proof There is a possibility that insiders leak user information at the gateway. In the user registration stage, the user's registered password PW_i is protected by $HPW_i = h(PW_i \| \alpha_i)$, and the insider may obtain HPW_i. Based on the unidirectional nature of the hash function, the insider is unable to compute PW_i by $HPW_i = h(PW_i \| \alpha_i)$[33]. In addition, HPW_i also contains the user's biometric α_i, and the insider cannot obtain α_i to guess the correct PW_i by offline guessing. Therefore, the scheme can resist internal attacks by setting HPW_i.

Proposition 6 *The scheme is resistant to tampering attacks.*

Proof Suppose the attacker tampers with the message sent by the user to the gateway, and the gateway receives the message and needs to verify whether $D_3^{*} = h(TID_i^{'} \| D_0^{*} \| C_u \| x_i \| K_i \| T_1)$ is equal to D_3. To crack D_3, the attacker needs to have both the user's private key r_u, identifier ID_i, password PW_i, secret x_i, and key parameter K_i[34], etc. The above parameters are not propagated in plaintext over the public channel, and the attacker cannot verify them through the gateway. Therefore, the scheme makes it impossible for an attacker to authenticate D_3 by setting multiple parameters. The scheme is resistant to tampering attacks.

Proposition 7 *The scheme is resistant to replay attacks.*

Proof A replay attack occurs when an attacker sends a packet that has been received by the target for the purpose of spoofing the system. All the messages sent in the two-way authentication process contain the timestamp T, and all parties need to verify whether the time difference is less than ΔT after receiving the message. If the attacker carries out replay attacks, the replayed message can be recognized by verifying the timestamp. The scheme resists replay attacks by adding timestamps.

	Xue et al.[16]	Mo et al.[39]	Deng et al.[40]	Meriam et al.[6]	Proposed Scheme
Year	2021	2022	2022	2022	2022
Forward security	×	×	×	×	√
Resist KSSTI attacks	×	×	×	√	√
Resist internal privilege attacks	√	×	√	×	√
Resist offline dictionary attacks	√	√	√	√	√
Clock synchronization	√	×	√	×	√
Anonymity	×	√	√	√	√
Resist MITM attacks	×	×	√	√	√
Resist user registration attacks	×	√	√	√	√

Table 3. Comparison of security features. Remarks: √: Yes ×: No.

Proposition 8 *The scheme is resistant to MITT attacks.*

Proof According to the challenge/response mechanism, both the user and the gateway or the sensor and the gateway need to verify each other's identity. According to Propositions 4 and 6, which have already been proven, the attacker cannot disguise the user or tamper with the message, so the attacker cannot launch a MITT attack disguised as an intermediary. The same can be proven for the communication between sensors and gateways. In addition, timestamps and random numbers are fresh and cannot be forged by an MITT attack[35]. Therefore, an attacker cannot disguise him- or herself as an MITT to launch an attack. The scheme makes it impossible for the attacker to accomplish MITT attacks by authenticating the user, gateway, and sensor.

Proposition 9 *The scheme is resistant to Denning-Sacco attacks*[36].

Proof Suppose the attacker steals the agreement key $SK = h(SID_j \| r_g \| R_{su} \| C_u \| C_s \| TID_i')$. SK is the hash function's hash value[37], and according to its one-way property, the attacker cannot obtain the parameters in SK. In addition, the parameters in SK such as user private key r_u, gateway private key r_g, sensor private key r_s, C_u, and C_s are not transmitted in the public channel, and the attacker cannot complete the Denning-Sacco attack.Therefore, the scheme resists Denning-Sacco attacks by performing hash transformations on the session key SK and by making SK have more complex parameters.

Proposition 10 *The scheme has forward security.*

Proof Assuming that the attacker intercepts the public keys R_u and R_s of the user and the sensor, the calculation of SK also requires r_u, r_g, r_s, C_u, and C_s. None of these parameters are transmitted in the public channel, and they cannot be obtained by the attacker. An attacker trying to calculate r_s and r_u by $R_s = r_s{}^*P$ and $R_u = r_s{}^*P$, or $r_s{}^*R_u$ and $R_s{}^*r_u$ by $R_s{}^*R_u$ cannot do so because the above computations involve ECCDLP mathematical puzzles. Therefore, the scheme is forward-safe.

Proposition 11 *The scheme enables both two-way authentication and key agreement.*

Proof The scheme through $D_3 = h(TID_i' \| D_0 \| C_u \| x_i \| K_i \| T_1)$ and $D_{14} = h(TID_i' \| x_i' \| K_i' \| r_g \| C_s \| SID_j \| D_0 \| T_4)$ achieves two-way authentication of the user and the gateway and through $D_7 = h(TID_i' \| SID_j \| C_u \| r_g \| x_j \| T_2)$ and $D_9 = h(SID_j \| r_g \| D_8 \| x_j \| C_s \| T_3)$ achieves two-way authentication of the gateway and the sensor, while the session key $SK_s = h(SID_j \| r_g \| R_{su} \| C_u \| C_s \| TID_i') = h(SID_j \| r_g \| R_{us} \| C_u \| C_s \| TID_i') = SK_u$ is negotiated during the authentication process.

Table 3 shows the security comparison of each scheme. It can be seen that this scheme has better security.

Efficiency analysis

The sensor nodes of WSNs have the characteristics of limited resources and low computation. In this section, they analyze the performance of scheme in analysed from two aspects—computation overhead and communication overhead—and the scheme is proven to be suitable for resource-constrained WSNs through comparisons with other schemes[38].

Computational overhead

The computational overhead is mainly considered for recovering biometric features, point multiplication, modular exponentiation, symmetric encryption/decryption, hashing, and so forth. The computational overhead of XOR and concatenation is very small and negligible compared to other operations. Referring to the literature[15], the computational elapsed time is shown in Table 4; the comparison of computational overheads of each scheme is shown in Table 5.

Notations	Descriptions	Time consuming (ms)
T_{FE}	Time of recover biometric features	1.989
T_{ecm}	Time of point multiplication operation	1.989
T_{mm}	Time of Modular exponentiation operation	0.171
$T_{E/D}$	Time of symmetric encryption/decryption operations	0.00325
T_h	Time of hash operation	0.0026

Table 4. The notations, descriptions, and time consuming required for computational time.

	U_i	GWN	S_j	合计
Xue et al.[16]	$13T_h+1T_{FE}$	$18T_h$	$6T_h$	$37T_h+1T_{FE}$
Mo et al.[39]	$2T_{ecm}+12T_h+1T_{FE}$	$10T_h+1T_{E/D}$	$2T_{ecm}+5T_h+1T_{E/D}$	$4T_{ecm}+27T_h+2T_{FE}+1T_{FE}$
Deng et al.[40]	$2T_{ecm}+14T_h+1T_{FE}$	$13T_h$	$2T_{ecm}+7T_h$	$4T_{ecm}+34T_h+1T_{FE}$
Meriam et al.[6]	$4T_{ecm}+8T_h+T_{E/D}$	$2T_{ecm}+5T_h+T_{E/D}$	$2T_{ecm}+2T_h$	$8T_{ecm}+15T_h+2T_{E/D}$
Proposed scheme	$5T_{ecm}+22T_h+1T_{FE}$	$4T_{ecm}+18T_h$	$3T_{ecm}+8T_h$	$12T_{ecm}+48T_h+1T_{FE}$

Table 5. Comparison of computational overhead.

From the computational time consumption in Table 4, it can be seen that the T_{FE} and T_{ecm} time consumption is high, and the T_{FE} of each scheme is similar, so the focus is on the point multiplication operation T_{ecm}. This scheme uses the ECC-based key agreement scheme, and the point multiplication operation overhead is higher than that of other schemes, but it has higher security compared to other schemes that only use hash computation or symmetric encryption and decryption schemes. WSNs focus on the computational overhead of resource-constrained sensor nodes. The computational overhead of the sensor nodes is increased only once compared to schemes[6,39], and[40], which also have point multiplication operations. This scheme does not put too much pressure on sensor computation. Although the other schemes have less computational overhead, the present scheme is more effective in dealing with various security threats and is more suitable for high security systems.

Communication overhead

The communication overhead is mainly for the data lengths of identity, hash value, fuzzy extractor public data, random numbers, timestamp, points of elliptic curve (public key), and symmetric encryption/decryption data. To facilitate the comparison, each data length in this scheme is set uniformly. The specific values are shown in Table 6, the comparison of communication overheads of each scheme is shown in Table 7, and the specific communication overhead quantization diagrams are shown in Figs. 12 and 13[41].

Notations	Descriptions	Length(bit)
L_{ID}	Identity length	32
L_h	Hash value length	160
L_{FE}	Fuzzy extractor public data length	128
L_r	Random number length	128
L_T	Timestamp length	32
L_{ECC}	Points of elliptic curve (public key) length	160
$L_{E/D}$	Symmetric encryption/decryption data length	128

Table 6. The notations, descriptions, and lengths required for communication data.

	U_i	GWN	S_j	Total
Xue et al.[16]	$3L_{ID}+1L_{FE}+6L_h+1L_T$	$1L_{ID}+1L_{FE}+11L_h+1L_T$	$1L_{ID}+2L_h+1L_T$	$5L_{ID}+2L_{FE}+19L_h+3L_T$
Mo et al.[39]	$1L_{ID}+7L_h+1L_T$	$1L_{ECC}+1L_{E/D}+1L_{FE}+5L_h+3L_T$	$1L_{ECC}+2L_h+1L_T$	$1L_{ID}+2L_{ECC}+1L_{E/D}+1L_{FE}+14L_h+5L_T$
Deng et al.[40]	$1L_{ECC}+5L_h$	$2L_{ECC}+10L_h$	$1L_{ECC}+2L_{FE}$	$4L_{ECC}+15L_h+2L_{FE}$
Meriam et al.[6]	$1L_{ECC}+4L_{E/D}+3L_h+1L_T$	$2L_{ECC}+2L_h+2L_T$	$1L_{ECC}+1L_h+1L_T$	$4L_{ECC}+4L_{E/D}+6L_h+4L_T$
Proposed scheme	$2L_{ECC}+4L_h+1L_T$	$3L_{ECC}+10L_h+2L_T$	$1L_{ECC}+2L_h+1L_T$	$6L_{ECC}+16L_h+4L_T$

Table 7. Communication overhead comparison.

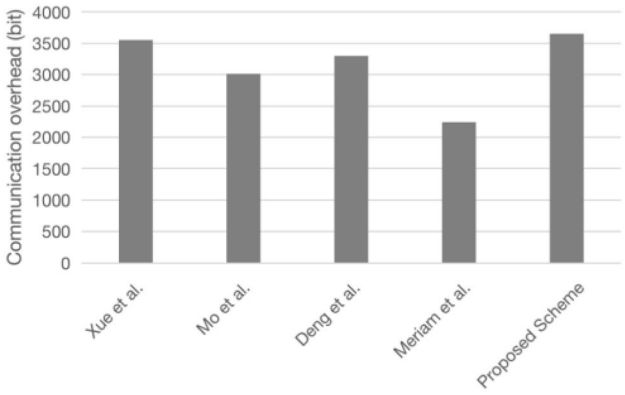

Figure 12. Total communication overhead comparison.

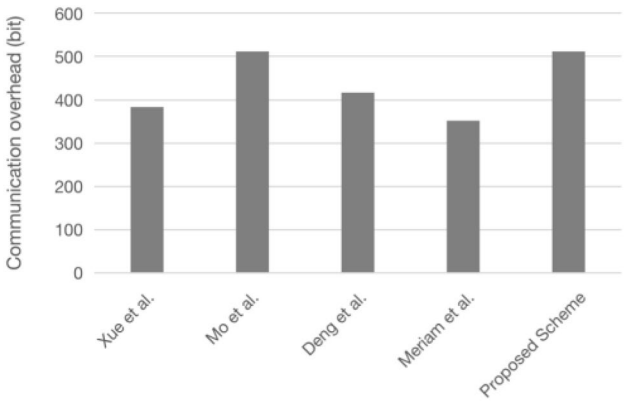

Figure 13. Comparison of node communication overhead.

This scheme is based on ECC, and as the communication process needs to send each party's public key several times, the communication overhead is slightly higher than with other schemes. For the communication overhead of resource-constrained sensor nodes, this scheme is the same as scheme[39] and slightly higher than schemes[6,16] and[40], but still within the tolerance range of sensor nodes and suitable for WSNs.

Conclusions

This paper examines multifactor authentication for WSNs. First, related schemes from recent years are introduced, and based on this, the scheme of Xue et al.[16] is examined, with a focus on its advantages and security vulnerabilities. Then, a three-factor authentication and key agreement scheme based on ECC is proposed for WSNs. The security of the scheme is demonstrated by the BAN logical and informal analysis, and efficiency analysis shows that the scheme is used for resource-constrained WSNs. Overall, the proposed scheme effectively improves the security performance of WSNs based on efficiency and has good application value. Due to the use of ECC dot-multiplication operations, the computational energy consumption of the scheme is still higher compared to the scheme with only hash operations; therefore, in the next step of this research, the efficiency of the scheme needs to be further improved to guarantee security.

Data availability

The authors confirm that the data supporting the findings of this study are available within the article and its supplementary materials.

Received: 4 September 2023; Accepted: 14 January 2024
Published online: 20 January 2024

References

1. Mishra, D. *et al.* Efficient authentication protocol for secure multimedia communications in IoT-enabled wireless sensor networks. *Multimed. Tools Appl.* **77**, 18295–18325 (2018).
2. Lee, Y. & Kim, H. Anonymous password-based authenticated key agreement scheme with non-tamper resistant smart cards. *Int. J. Secur. Appl.* **9**(11), 419–428 (2015).

3. Wu, M., Chen, J. & Wang, R. An enhanced anonymous password-based authenticated key agreement scheme with formal proof. *Int. J. Netw. Secur.* **19**(5), 785–793 (2017).
4. Jiang, Q. *et al.* An untraceable temporal-credential-based two-factor authentication scheme using ECC for wireless sensor networks. *J. Netw. Comput. Appl.* **76**, 37–48 (2016).
5. Li, X. *et al.* A three-factor anonymous authentication scheme for wireless sensor networks in internet of things nvironments. *J. Netw. Comput. Appl.* **103**, 194–204 (2018).
6. Meriam, F., Hassan, E. G. & Ahmed, T. A lightweight ECC-based three-factor mutual authentication and key agreement protocol for WSNs in IoT. *Int. J. Adv. Comput. Sci. Appl (IJACSA)* **13**(6), 491–501 (2022).
7. Wu, F., Xu, L., Kumari, S. & Li, X. A privacy-preserving and provable user authentication scheme for wireless sensor networks based on Internet of Things security. *J. Ambient Intell. Humaniz. Comput.* **8**(1), 101–116 (2017).
8. Wu, F. *et al.* An efficient authentication and key agreement scheme for multi-gateway wireless sensor networks in IoT deployment. *J. Netw. Comput. Appl.* **89**, 72–85 (2016).
9. Bayat, M., Atashgah, M. B., Barari, M. & Aref, M. R. Cryptanalysis and improvement of a user authentication scheme for internet of things using elliptic curve cryptography. *Int. J. Netw. Secur.* **21**(6), 897–911 (2019).
10. Guo, H., Gao, Y., Xu, T., Zhang, X. & Ye, J. A secure and efficient three-factor multi-gateway authentication protocol for wireless sensor networks. *Ad Hoc Netw.* **95**, 101965 (2019).
11. Jung, J., Moon, J., Lee, D., Won, D. & Akkaya, K. Efficient and security enhanced anonymous authentication with key agreement scheme in wireless sensor networks. *Sensors* **17**(3), 644 (2017).
12. Sravani, C. *et al.* Secure signature-based authenticated key establishment scheme for future iot applications. *IEEE Access* **5**, 3028–3043 (2017).
13. Singh, M. & Mishra, D. Post-quantum secure authenticated key agreement protocol for wireless sensor networks. *Telecommun. Syst.* **84**(1), 101–113 (2023).
14. Azrour, M., Mabrouki, J., Guezzaz, A. & Farhaoui, Y. New enhanced authentication protocol for internet of things. *Big Data Min. Anal.* **4**(1), 1–9 (2021).
15. Vinoth, R., Deborah, L. J., Vijayakumar, P. & Kumar, N. Secure multifactor authenticated key agreement scheme for industrial IoT. *IEEE Internet Things J.* **8**(5), 3801–3811 (2021).
16. Xue, L., Huang, Q., Zhang, S., Huang, H. & Wang, W. A lightweight three-factor authentication and key agreement scheme for multigateway WSNs in IoT. *Secur. Commun. Netw.* **2021**, 1–15 (2021).
17. Liu, Z., Li, Z., Zhang, Q., Dong, S., Liu, J. & Zhao, Y. Two-factor authentication and key agreement schemes for smart home fingerprint characteristics. *Mobile Inf. Syst.* (2022).
18. Srinivas, J., Mishra, D., Mukhopadhyay, S. & Kumari, S. Provably secure biometric based authentication and key agreement protocol for wireless sensor networks. *J. Ambient. Intell. Humaniz. Comput.* **9**, 875–895 (2018).
19. Liu, S. M., Ye, J. Y. & Wang, Y. L. Improvement and security analysis on symmetric key authentication protocol Needham-Schroeder. *Appl. Mech. Mater.* **513**, 1289–1293 (2014).
20. Lai, C., Ma, Y., Lu, R., Zhang, Y. & Zheng, D. A novel authentication scheme supporting multiple user access for 5g and beyond. *IEEE Trans. Depend. Secure Comput.* **2022**, 1–16 (2022).
21. Yang, Y., Zheng, X., Guo, W., Liu, X. & Chang, V. Privacy-preserving fusion of IoT and big data for e-health. *Future Gener. Comput. Syst.* **86**, 1437–1455 (2018).
22. Tyagi, P., Kumari, S., Alzahrani, B. A., Gupta, A. & Yang, M. H. An enhanced user authentication and key agreement scheme for wireless sensor networks tailored for IoT. *Sensors* **22**, 8793 (2022).
23. Liu, S., Li, X., Wu, F., Liao, J. & Lin, D. A novel authentication protocol with strong security for roaming service in global mobile networks. *Electronics* **8**(9), 939 (2019).
24. Ansari, A. A., Gera, P., Mishra, B. & Mishra, D. A secure authentication framework for WSN-based safety monitoring in coal mines. *Sādhanā* **45**, 1–16 (2020).
25. Chen, C. M., Liu, S., Chaudhry, S. A., Chen, Y. C. & Khan, M. A. A lightweight and robust user authentication protocol with user anonymity for IoT-based healthcare. *Comput. Model. Eng. Sci.* **130**(4), 307–329 (2022).
26. Chen, Y., López, L., Martínez, J. F.&Castillejo, P. A lightweight privacy protection user authentication and key agreement scheme tailored for the internet of things environment: Lightpriauth. *J. Sens.* (2018).
27. Guo, J. & Du, Y. A secure three-factor anonymous roaming authentication protocol using ECC for space information networks. *Peer Peer Netw. Appl.* **14**(2), 898–916 (2021).
28. Sani, S. A., Dong, Y., Yeoh, P. L., Wei, B. & Vucetic, B. A lightweight security and privacy-enhancing key establishment for internet of things applications. In*2018 IEEE International Conference on Communications (ICC)* (2018).
29. Boyko, V., MacKenzie, P. & Patel, S. Provably secure password-authenticated key exchange using Diffie-Hellman. In *Advances in Cryptology—EUROCRYPT 2000: International Conference on the Theory and Application of Cryptographic Techniques Bruges, Belgium, 14–18 May, 2000 Proceedings 19* 156–171 (Springer, 2000).
30. Mo, J. & Chen, H. A lightweight secure user authentication and key agreement protocol for wireless sensor networks. *Secur. Commun. Netw.* **2019**, 1–17 (2019).
31. Zhou, Z., Wang, P. & Li, Z. A quadratic residue-based RFID authentication protocol with enhanced security for TMIS. *J. Ambient Intell. Humaniz. Comput.* **10**(9), 3603–3615 (2019).
32. Kumar, D., Grover, H. S., Kaur, D., Verma, A. & Kumar, B. An efficient anonymous user authentication and key agreement protocol for wireless sensor networks. *Int. J. Commun. Syst.* **34**(5), e4724 (2021).
33. Kamil, I. A. & Ogundoyin, S. O. A lightweight mutual authentication and key greement protocol for remote surgery application in Tactile Internet environment. *Comput. Commun.* **170**, 1–18 (2021).
34. Khalid, H., Hashim, S. J., Ahmad, S. M. S., Hashim, F. & Chaudhary, M. A. Robust multi-gateway authentication scheme for agriculture wireless sensor network in society 5.0 smart communities. *Agriculture* **11**(10), 1020 (2021).
35. Alharbi, M. H. & Alhazmi, O. H. User authentication scheme for internet of things using near field communication. *Int. J. Reliab. Qual. Saf. Eng.* **27**(5), 2040012 (2020).
36. Abbas, G., Tanveer, M., Abbas, Z. H., Waqas, M. & Baker, T. A secure remote user authentication scheme for 6LoWPAN-based Internet of Things. *Plos One* **16**(11), e0258279 (2021).
37. Yeh, H., Chen, T., Liu, P., Kim, T. & Wei, H. A secured authentication protocol for wireless sensor networks using elliptic curves cryptography. *Sensors* **11**, 4767–4779 (2011).
38. Zhang, S., Du, X. & Liu, X. An efficient and provable multifactor mutual authentication protocol for multigateway wireless sensor networks. *Secur. Commun. Netw.* **2021**, 1–17 (2021).
39. Mo, J., Hu, Z. & Shen, W. A provably secure three-factor authentication protocol based on chebyshev chaotic mapping for wireless sensor network. *IEEE Access* **10**, 12137–12152 (2022).
40. Deng, D. Research on key technologies of authentication and secret key management based on non-traditional certificates in WSN. *Univ. Electron. Sci. Technol.* (2022).
41. Jo, H. R., Pak, K. S., Kim, C. H. & Zhang, I. J. Cryptanalysis and improved mutual authentication key agreement protocol using pseudo-identity. *Plos One* **17**(7), e0271817 (2022).

Acknowledgements
The author would like to thank his esteemed college leaders for their support of his scientific work and his colleagues Yanwu Di, Si Chen, Chaoyang Huang, Wenliang Liu, and Yan Li for their contributions to this work. This work was supported by the research project "Research on Key Management Scheme for Clustered Wireless Sensor Networks" (Grant Number [KYZ202203]) of Xiamen Ocean Vocational College (China).

Author contributions
W.H. independently conceived and designed the study; prepared the materials, collected and analyzed the data; and wrote the first and final drafts of the manuscript.

Competing interests
The author declares no competing interests.

Additional information
Correspondence and requests for materials should be addressed to W.H.

Reprints and permissions information is available at www.nature.com/reprints.

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.