Online-Journals.org

iJOE International Journal of Online and Biomedical Engineering

iJOE | eISSN: 2626-8493 | Vol. 21 No. 1 (2025) | OPEN ACCESS

https://doi.org/10.3991/ijoe.v21i01.51633

PAPER

ECC-Based Anonymous and Multi-factor Authentication Scheme for IoT Environment

Souhayla Dargaoui[1], Mourade Azrour[1], Ahmad El Allaoui[1], Azidine Guezzaz[2], Abdulatif Alabdulatif[3](✉), Sultan Ahmad[4]

[1]Faculty of Sciences and Techniques, Moulay Ismail University of Meknes, Errachidia, Morocco

[2]Higher School of Technology, Cadi Ayyad University, Essaouira, Morocco

[3]Department of Computer Science, College of Computer, Qassim University, Buraydah, Saudi Arabia

[4]College of Computer Engineering and Sciences, Prince Sattam Bin Abdulaziz University, Al-Kharj, Saudi Arabia

ab.alabdulatif@qu.edu.sa

ABSTRACT

Owing to its capability to offer remote services, the Internet of Things (IoT) has immersed itself in all areas of our daily lives. However, this big use of IoT networks makes the user's data change insecurely in open channels vulnerable to malicious use. As a result, the security of the user's data in an IoT environment becomes a critical issue. Given that authentication is a mechanism that may prevent hackers from retrieving and exploiting data communicated between IoT devices, researchers have proposed many lightweight IoT authentication schemes in the last decades. However, most of these schemes are based on two authentication factors and are unable to ensure unlink ability, key secrecy, perfect forward secrecy, and resistance to node capture, denial of service (DoS) attacks, stolen verifiers, denning-SSACO attacks, and GWN bypassing. In this paper, we present an anonymous three-factor authentication scheme based on elliptic curve cryptography (ECC), which can provide all security services and resist well-known attacks. Then, based on informal security analysis and the formal security proof using ProVerif we show that our provided scheme is secure and can resist known attacks. Finally, we show the comparison result among our protocol and other protocols in terms of computation overheads, communication overheads, and security features.

KEYWORDS

authentication, Internet of Things (IoT), elliptic curve cryptography (ECC), three-factor security, biometric

1 INTRODUCTION

Recently, the Internet of Things (IoT), as a large network that interlinked goods, devices, and databases, has facilitated human daily life, providing remote management of aspects including transportation, healthcare, energy, smart buildings, and the surroundings [1], [2], [3], [4], [5], [6], [7], [8], [9], [10], [11]. Balasundaram et al. [12] afford a smart healthcare diagnostics system that merges the IoT with recurrent neural networks. The object of the presented mechanism is to classify health anomalies accurately. To manage car parks in a smart city, Amara Aditya et al. [13] present an

Dargaoui, S., Azrour, M., El Allaoui, A., Guezzaz, A., Alabdulatif, A., Ahmad, S. (2025). ECC-Based Anonymous and Multi-factor Authentication Scheme for IoT Environment. *International Journal of Online and Biomedical Engineering (iJOE)*, 21(1), pp. 56–75. https://doi.org/10.3991/ijoe.v21i01.51633

Article submitted 2024-08-07. Revision uploaded 2024-10-16. Final acceptance 2024-10-16.

© 2025 by the authors of this article. Published under CC-BY.

intelligent car park based on IoT. They provide a framework that first assembles real-time data, then analyzes this data, and provides the coordinates of an available place at a nearby location. Using this immersed technology, users generate and exchange an important data flow every day using insecure wireless communication networks. As a result, this data becomes susceptible to illegitimate exploitation. Several solutions may be placed in an IoT network to protect user's data from malicious use. The most important and efficient solution is authentication.

Authentication, generally, prevents the hacker from exploiting a customer's data even if it is extracted from a transmitted message; at the same time, it allows legitimate entities to use this data freely and securely. Unfortunately, traditional authentication schemes require high computational power, storage memory, and energy, things that cannot be ensured by IoT devices known for their limitations. As a result, lightweight authentication protocols have been immersed. During the previous ten years, many lightweight authentication protocols have been proposed [14], [15], [16], [17], [18], [19], [20], [21], [22], [23], [24], [25], [26], [27], [28], [29], [30], [31], [32], [33], [34], [35], [36], [37]. However, the analysis shows that most of these protocols are two-factor schemes and cannot provide unlink ability, key secrecy, and perfect forward secrecy, nor resist node capture, denial of service (DoS) attack, stolen verifier, denning-SSACO attack, and GWN bypassing.

Our article intends to provide a lightweight authentication and key accordance scheme, which improves the robustness against well-known attacks and ensures the required security services. The contributions under the proposed strategy are as follows:

- We present a novel three-factor authentication and key accordance scheme using a smart card, biometrics, and a password. The provided scheme will present two reciprocal authentications: firstly, between the customer and the portal, and secondly, between the portal and the smart device. Finally, we will establish a session key between the customer and the device.
- The offered protocol is built on a one-way hash function, elliptic curve cryptography (ECC), and random numbers.
- Informal security analysis and the simulation using ProVerif show the efficiency of the afforded scheme and its robustness.

The remaining part of our paper is structured as follows: Section 2 presents linked works. In Section 3, the provided protocol is presented. The informal and formal examinations of security are presented in Section 4. Section 5 provides a comparative analysis between our system and some others concerning calculation overheads, communication costs, and security features. In Section 6, we conclude the paper.

2 RELATED WORKS

Over the last few years, numerous authentication and key accords schemas were suggested to guarantee secrecy and protection in IoT environments. Given that the symmetric encryption process is very fast, needs short keys, provides an encoded text of the same size or smaller than the original ordinary text, and requires fewer resources compared with other mechanisms. It seems to be the most suitable mechanism for implementing IoT authentications. P. Gope and T. Hwang [38] introduced a realistic authentication protocol helpful in WSN that may secure user anonymity, untraceability, and onward/backward secrecy. A. Ghani et al. [39] presented a cryptanalysis of [38], and they confirmed that it is insecure upon user tracking stolen verifier and DoS. More than that, A. Ghani et al. provided an improved symmetrical key authentication scheme for IoT-based WSNs, and they demonstrated the ability of their protocol to remedy the Gope and Hwang scheme's weaknesses. The analysis of the comparison result between their scheme and [38] shows that they have the same communication cost. However, computational cost [39] has 52.63% effectiveness over the basic schema.

Sadly, symmetric encryption has some weaknesses compared to public key encryption mechanisms, which provide confidentiality, authenticity, and non-repudiation. Recently, many researchers have conceived IoT authentication schemes based on asymmetric encryption. D.Q. Bala et al. [40] presented an IoT authentication protocol that uses the CL-PKC technique. They proved that the suggested schema was robust regarding node impersonation and replay attacks. N. Li et al. [41] proposed a lightweight bi-directional authentication scheme for smart city applications that is built on public key encryption. The provided protocol balances the effectiveness and communication costs without compromising security. N. Li et al. proved that their protocol was more performant than available protocols at that epoch.

Compared to public key cryptosystems such as RSA, El Gamel and quadratic-based public cryptosystems, ECC demands a much shorter key length but still offers the same security strength. Q. Jiang et al. [42] demonstrated that D. He et al.'s scheme [43] is impressionable to harmful user imitation attacks and stolen smart-card assaults. On the other hand, they illustrated that He et al.'s authentication scheme could not ensure untraceability and was sensitive to traceability threats. They then came up with a two-factor authentication scheme founded upon temporal-credential exploiting the ECC for WSN, which provides the missing security features while preserving the desired characteristics of the baseline scheme. Li et al. [44] reviewed Q. Jiang et al.'s scheme and offered another one that is based on three factors. The result of the performance comparison displays that their scheme affords more security features, all while keeping the same computational efficiency.

For more improvement, many other techniques were used in IoT authentication. M. A. Qureshi [45] proposed PUF-IPA, an authentication system using physically unclonable functions that ensure bolstered resistance over security threats in comparison with earlier protocols based on the same basics. Results analysis shows the robustness of PUF-IPA. M. T. Hammi [46] presented an unfocused IoT authentication scheme using blockchain, which ensures a robust certification of IoT devices. Using C++ language and the Ethereum blockchain, the proposed protocol was implemented, and the results show its efficiency and low cost.

3 MATERIALS AND METHODS

In this section, we outline the steps followed to construct our authentication protocol. Generally, there are three steps: review study, classification and cryptography method selection, and implementation and examination.

The review was an exhaustive study of some authentication schemes published from 2019 to 2024. This study explored authentication schemes focusing on cryptography techniques and authentication factor numbers. In this step, we have defined some research directions that need special interest to overcome the gap in existing IoT authentication and the cryptography methods most used in authentication: ECC, blockchain, and hash functions.

In the second step, we studied the characteristics of each cryptography method. As a result, blockchain-based authentication can provide high-security performance. However, it requires high energy, high computational cost, and expensive storage overhead. On the other hand, ECC can also allow high security and low latency, using less computation power and memory resources. While hash function-based authentication requires less computer power and does not provide a high-security level. Finally, we choose ECC because of the limited nature of IoT devices.

In the third step, we developed our authentication scheme, as explained in Section 4. Then, we evaluated the proposed scheme using the AVISPA tool, as demonstrated in Section 5. Finally, we examined the scheme based on security features and computation and communication costs, as explained in Section 6.

4 RESULTS: THE PROPOSED SCHEME

In that part, we provide an anonymous and multi-factor IoT authentication scheme using ECC. Our mechanism includes five stages, which are the initialization stage, the sensor addition stage, the registration stage, the login and authentication stage, and the password change stage. Figure 1 illustrates the network model and the different stages. Table 1 presents the required notations.

Table 1. Notations

Notation	Description
U	User
S	Sensor
ID_i	User identity
SID_j	Sensor identity
PW_i	User password
GWN	Gateway
K_G	Gateway secret key
h (.)	One-way hashing function
SK	Session key
Gen (.)	Fuzzy extractor generation procedure
Rep (.)	Fuzzy extractor reproduction procedure
K_u, R_u, K_s, R_s, R_g, R_G, r1	Random numbers
T_i	Timestamp
Bio_i	User biometric model
⊕	Xor procedure
||	String concatenation procedure
P	The generator points on the curve
ΔT	Maximum transmission delay

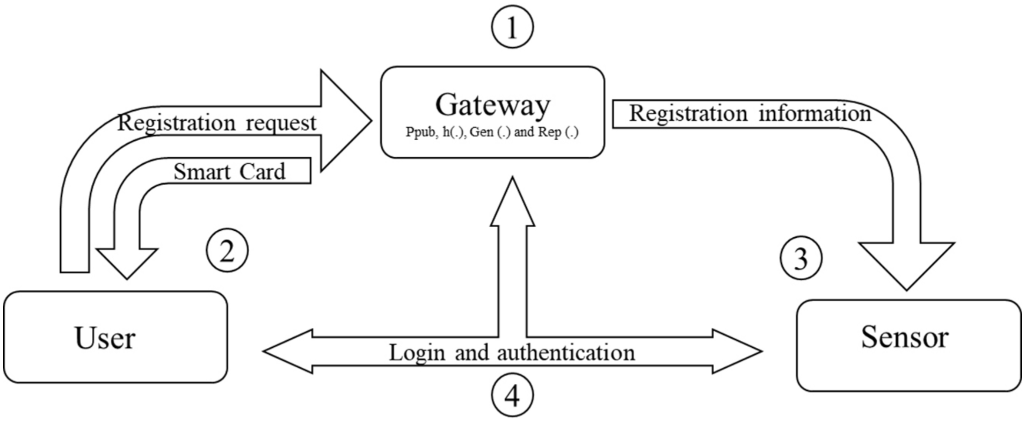


Fig. 1. System model

4.1 Initialization phase

In the initialization stage, the gateway (GWN) chooses an arbitrary number KG as the private key, chooses P, an elliptical curve generator points and a warrant hashing function, and calculates Ppub = KG. P as the GWN public key. Then, it publishes these elements with the generation and reproduction algorithms of the fuzzy extractors Gen (.) and Rep (.).

4.2 New sensor addition phase

In this stage, the Gateway produces an identity SIDj for the sensor, stores this identity in its database, then calculates the value C = h (SIDj||KG), and then it sends {SIDj, C, P} to the sensor, which stores them in its storage.

4.3 User registration phase

To complete the enrollment stage, the user must go through three steps:

First step: the user U adopts his identity IDi and keyword PWi, scans his fingerprints, and generates a random number r1 to calculate the pseudo identifier HID = h (IDi||r1), then he calculates H = HID ⊕ h (IDi||PWi) to hide HID. Then, using a fuzzy extractor, it generates Ri and Pi, (Ri, Pi) = Gen (Bioi), and it calculates HPW = h (PWi||Ri) and W = HPW ⊕ h (IDi||PWi). Thereafter, the user communicates HID and HPW to the gateway.

Second step: the Gateway calculates A = h (HID||KG|HPW) and sends it to the user after storing HID in its database.

Third step: the user calculates B = A ⊕ h (IDi||PWi) and stores {H, Pi, W, B, Rep (.), h (.), P} in a smart card.

4.4 Login and authentication phase

In this phase, the contact between the user, the portal, and the IoT device is established by an open broadcaster. The procedures of this phase are pictured in Figure 2, which illustrate the parameters stored by each entity and are further detailed below.

First, after inserting the smart card, entering the user IDi and password PWi, and scanning the fingerprints Bioi, the user reconstructs Ri using the Rep algorithm; Ri = Rep (Bio_i, Pi), then computes HPW = h (PWi||Ri) and HPW* = W ⊕ h (IDi||PWi) and checks if HPW = HPW*. If the values are identical, it generates T_1, Ku, Ru, and it calculates HID = H ⊕ h (IDi||PWi), A = B ⊕ h (IDi||PWi), M1 = h (A||T1||Ru), M2 = Ku.P, and M3 = h (Ku.Ppub) ⊕ (HID||HPW||Ru||SIDj). Finally, it sends the message {M1, M2, M3, T1} to the gateway.

After getting the user message, the gateway generates T2, checks the freshness of T1, recovers HID, HPW, Ru, and SIDj from M3 as illustrated in Figure 2 and checks the following equality: M1 = h (h (HID||KG||HPW)||T1||Ru). If the equality is true, the gateway generates Rg and RG and calculates M4 and M5, as depicted in Figure 2. Finally, it sends the message {M4, M5, RG, and T2} to the IoT device.

Then, right when the sensor receives the message sent by the gateway, it generates T3, verifies the validity of T2, and then it recovers Ru, Rg, and HID from the M5, it checks the next equality M4 = h(C||T2||Rg||HID). If the equality is true, it generates ks and Rs, and it calculates SK = h (Ru||Rg||Rs), M6, M7, and M8. At the end, the sensor sends the message {M6, M7, M8, T3} to the gateway.

When the gateway gets the sensor message, it generates T4, checks the freshness of T3, recovers Rs and SK from M8, and checks the following equality: M7 = h (h (SIDj||KG)||T3||Rs||HID). If the equality is true, it calculates M9 and M10. Ultimately, it transmits the message {M9, M10, T4} to the user.

During the last step, afterward, obtaining the gateway message, the user generates T5, tests the freshness of T4, recovers Rs, Rg, and SK from M9, and checks the following equalities: SK = h (Ru||Rg||Rs) and M1 = h (SK||HID||Rg|T4). If these equalities are true, authentication is complete, and the session key between the user and the IoT device is SK.

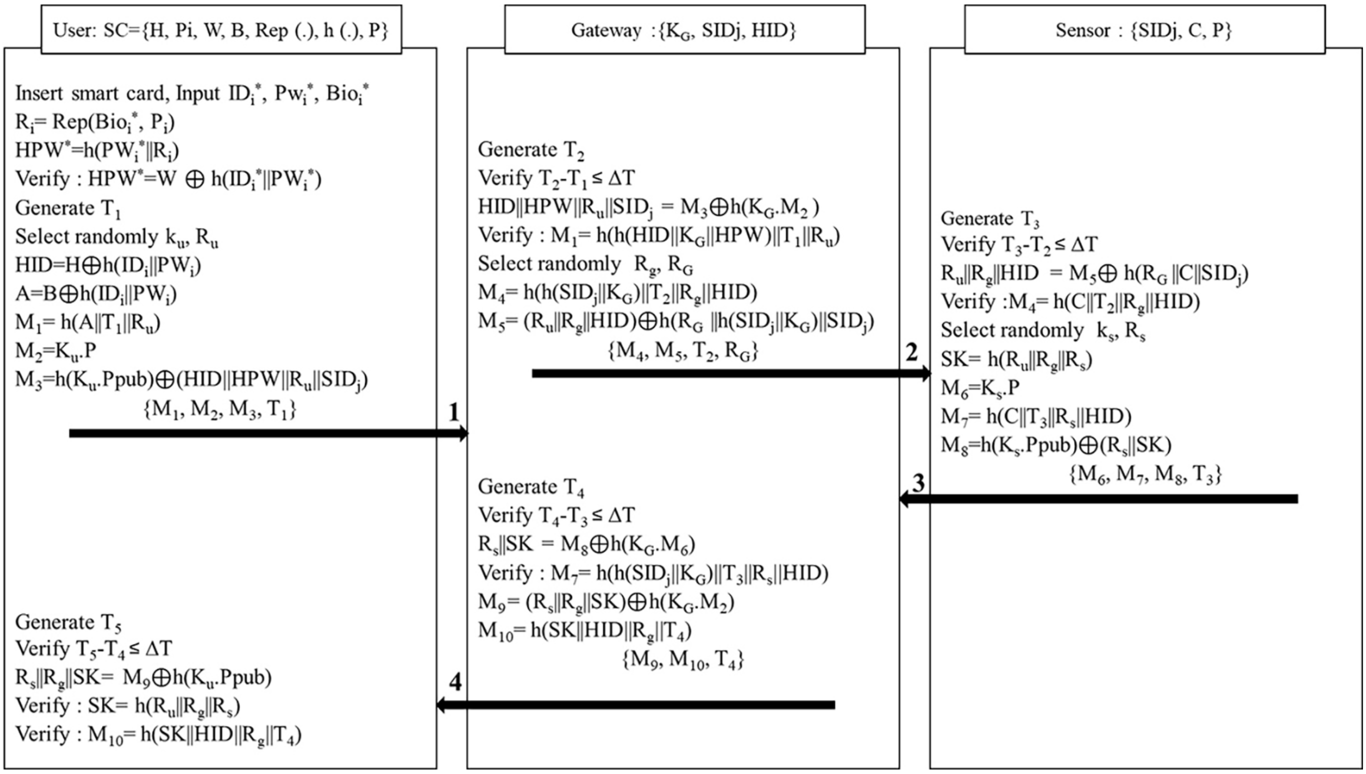
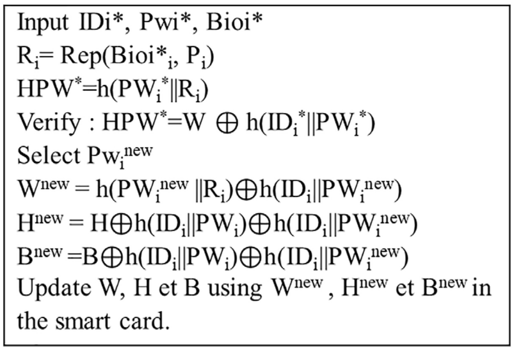

User: SC={H, Pi, W, B, Rep (.), h (.), P}
Insert smart card, Input ID_i^*, Pw_i^*, Bio_i^*
R_i= Rep(Bio_i^*, P_i)
HPW^*=h(PW_i^*||R_i)
Verify : HPW^*=W ⊕ h(ID_i^*||PW_i^*)
Generate T_1
Select randomly k_u, R_u
HID=H⊕h(ID_i||PW_i)
A=B⊕h(ID_i||PW_i)
M_1= h(A||T_1||R_u)
M_2=K_u.P
M_3=h(K_u.Ppub)⊕(HID||HPW||R_u||SID_j)
{M_1, M_2, M_3, T_1}
1
Gateway :{K_G, SIDj, HID}
Generate T_2
Verify T_2-T_1 ≤ ΔT
HID||HPW||R_u||SID_j = M_3⊕h(K_G.M_2)
Verify : M_1= h(h(HID||K_G||HPW)||T_1||R_u)
Select randomly R_g, R_G
M_4= h(h(SID_j||K_G)||T_2||R_g||HID)
M_5= (R_u||R_g||HID)⊕h(R_G ||h(SID_j||K_G)||SID_j)
{M_4, M_5, T_2, R_G}
2
Sensor : {SIDj, C, P}
Generate T_3
Verify T_3-T_2 ≤ ΔT
R_u||R_g||HID = M_5⊕ h(R_G ||C||SID_j)
Verify :M_4= h(C||T_2||R_g||HID)
Select randomly k_s, R_s
SK= h(R_u||R_g||R_s)
M_6=K_s.P
M_7= h(C||T_3||R_s||HID)
M_8=h(K_s.Ppub)⊕(R_s||SK)
{M_6, M_7, M_8, T_3}
3
Generate T_4
Verify T_4-T_3 ≤ ΔT
R_s||SK = M_8⊕h(K_G.M_6)
Verify : M_7= h(h(SID_j||K_G)||T_3||R_s||HID)
M_9= (R_s||R_g||SK)⊕h(K_G.M_2)
M_{10}= h(SK||HID||R_g||T_4)
{M_9, M_{10}, T_4}
4
Generate T_5
Verify T_5-T_4 ≤ ΔT
R_s||R_g||SK= M_9⊕h(K_u.Ppub)
Verify : SK= h(R_u||R_g||R_s)
Verify : M_{10}= h(SK||HID||R_g||T_4)

Fig. 2. Login and authentication phase

4.5 Password update phase

In this stage, the user first enters his username and password, then recovers Ri using the Rep algorithm Ri = Rep (Bioi, Pi), then calculates HPW = h (PWi||Ri) and HPW* = W ⊕ h (IDi||PWi) and checks if HPW = HPW*. If the values are identical, the user provides his new password, and then he calculates the new values of W, H, and B, as mentioned in Figure 3. Finally, it replaces the values of W, H, and B in the smart card with the up-to-date values.



Input IDi*, Pwi*, Bioi*
R_i= Rep(Bioi*$_i$, P_i)
$HPW^*=h(PW_i^*||R_i)$
Verify : $HPW^*=W \oplus h(ID_i^*||PW_i^*)$
Select Pw_i^{new}
$W^{new} = h(PW_i^{new}||R_i) \oplus h(ID_i||PW_i^{new})$
$H^{new} = H \oplus h(ID_i||PW_i) \oplus h(ID_i||PW_i^{new})$
$B^{new} = B \oplus h(ID_i||PW_i) \oplus h(ID_i||PW_i^{new})$
Update W, H et B using W^{new}, H^{new} et B^{new} in the smart card.

Fig. 3. Password update phase

5 SECURITY ANALYSIS

About the Dolev-Yao threat paradigm [47], we describe the capabilities of a hacker-like way:

- The attacker can spy on all communicated information via a public canal.
- The attacker may alter, add, replay, and redirect spyware messages.
- If the attacker gets the smart card from a customer, he may obtain all the information stored in this chip.
- If the hacker captures a device, he can get all the data kept in this device's memory.
- The adversary could be a legal user.

5.1 Informal security examination

Mutual authentication. In the provided protocol, the gate authenticates the user by comparing M1 with h (h (HID|KG||HPW)||T1||Ru). Then, the sensor authenticates the gateway by checking the following equality: M4 = h (C||T2||Rg||HID). Then, the gateway authenticates the sensor by evaluating the following equality: M7 = h (h (SIDj||KG)||T3||Rs||HID). Finally, the user authenticates the gateway by ensuring the following equality: M10 = h (SK||HID||Rg||T4). Hence, our proposed mechanism offers mutual authentication.

Anonymity and untrace ability. In our protocol, HID = h (IDi||r1) is included in M1, M3, M4, M5, and M7. According to the attack model, the opponent can have these messages. But it cannot extract HID from either M1, M4, or M7 because of the hash function or from M3 and M5 for reasons of the insolubility of the Computational Diffie–Hellman and the gateway secret key, respectively. In addition, even if it could know HID, it cannot extract IDi since it is hidden using r1 and function h (.). Therefore, the introduced protocol responds to the security standards for anonymity and untraceability.

Key security. Session key privacy requires that at the close of the authentication and key agreement phase, no one can recognize the session key except the customer and the gate. In our diagram, the session key is obtained as follows: SK = h (Ru||Rg||Rs). GWN is a trusted node, so the attacker cannot know KG. So even though an opponent may discover all the hidden data of the user Ku. P and Ks. P, he cannot calculate Ru, Rg, and Rs due to the untraceability of the CDH problem. On top of that, these values are random and change from one session to another. For these reasons, the protocol presented provides for the secrecy of session keys.

Impersonation attack. To impersonate the consumer, the pirate must calculate the message {M1, M2, M3, T1}, which is impossible without having HID, HPW, and A. Assuming that the attacker may steal the smart card, he cannot extract HPW from W = HPW ⊕ h (IDi||PWi), nor A from B = A ⊕ h (IDi||PWi), nor HID from H = HID ⊕ h (IDi||PWi) since it has not the user's login and password.

To usurp the identity of the GWN, the attacker must calculate the messages {M4, M5, T2, RG} and {M9, M10, T4}; for this, it must first extract HID, Ru, and SIDj from M3, which is impossible without having the secret key KG of the GWN.

Finally, the impersonation of the sensor requires the calculation of the message {M6, M7, M8, T3}, which is impossible without C and RG. Therefore, this type of attack does not exist in our scheme.

Replay attack. Suppose an opponent intercepts and replays the message {M1, M2, M3, T1}; the repeated message couldn't pass the GWN validation procedure if the timestamp is not valid. In addition, even if the opponent tries to modify T1 in the authentication demand, he couldn't modify the value of M1 without knowing A and Ru. The same goes for messages generated by the GWN and the sensor, so this kind of raid is impossible in our scheme.

Node capture. Capturing a sensor allows the attacker to know SIDj, C, and P, but because of the hashing function, the hacker may not discover KG. As such, capturing a sensor will not impact other sensors.

Denial of service. Because of the use of random values and timestamps, the attacker cannot trace the user by recording messages {M1, M2, M3, T1} because these message's arguments adjust by session variation, so he cannot change these messages. Therefore, this threat has no influence on our protocol.

Insider attack. This type of attack can occur when a legitimate (authenticated) user steals the password to use it to make another login request. In the proposed protocol, we used a hidden form of password HPW = h (PWi||Ri) where (Ri, Pi) = Gen (Bioi). So even if the attacker could find HPW, he can't recover PWi because of the hash function and Ri. For this reason, there are no internal threats in our protocol.

Stolen verifier. The portal doesn't contain any data corresponding to the checking table. Consequently, this attack does not exist in our diagram.

Denning-SSACO. This type of attack indicates the capability to extract a long-term secret key, such as a keyword, portal confidential key, or session key, from the previous session key. For the suggested protocol, this attack is impossible since the session key is calculated using random values and does not contain any long-term keys.

Smart card loss. As already cited in the previous paragraphs, the data kept in the chip card does not allow an adversary to go through verification without knowing IDi, PWi, and Bioi.

Password guessing. In this mechanism, M1, M2, and M3 are communicated using a public canal; even when an opponent spies on the conversation and gets the messages, he may not infer the keyword. To find HPW, it is necessary to solve the CDH problem. In addition, it cannot extract PWi from HPW.

On the other hand, once the data saved in the smart card has been extracted, the user can get B = A ⊕ h (IDi||PWi) and W = HPW ⊕ h (IDi||PWi). However, to find PWi, it must have IDi and Ri, which is impossible. That is why this type of assault cannot impact our system.

GWN Bypassing. We talk about a gate bypass assault once a legal, nonetheless, noxious consumer or a hacker may pass the verification stage without notifying the portal to complete its task. To do this, it must transmit a message {M4, M5, T2, RG} where M4 = h (h (SIDj||KG)||T2||Rg||HID), M5 = (Ru||Rg||HID) ⊕ h(RG || h(SIDj||KG)), correct to the Sj sensor, which is impossible without knowing the private gateway key. Therefore, this attack is not possible in our protocol.

The man in the middle. Imagine an opponent can intercept a legitimate connection demand {M1, M2, M3, T1}. He cannot falsify this message since he does not know Ru, Ku, IDi, and PWi. In conclusion, our protocol prevents MITM attacks.

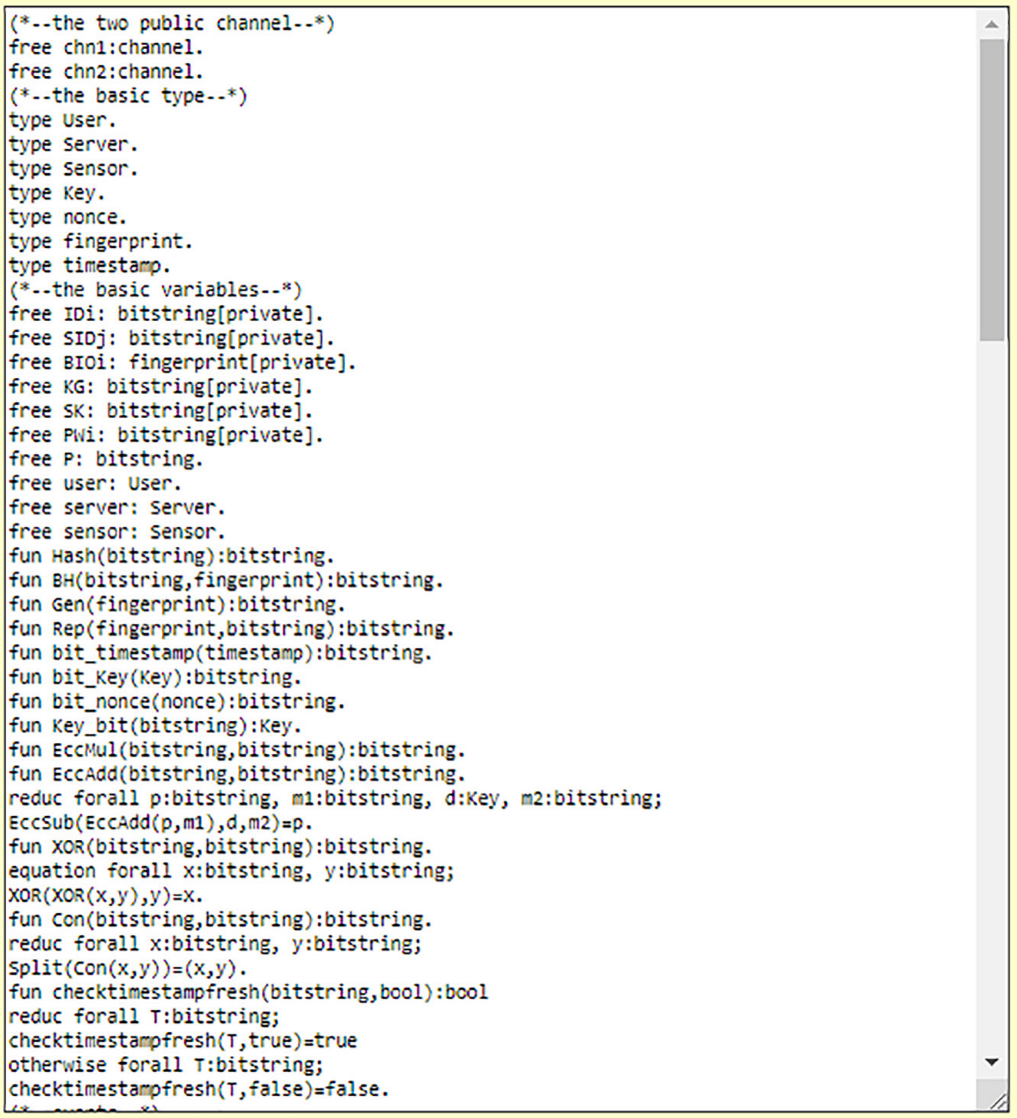
5.2 Formal security examination

In this part, the start is by explaining the usefulness of the ProVerif tool that is employed to formally analyze the security of the provided scheme. Next, we discuss the attained outcomes of this simulator. ProVerif is an automatic checker known for cryptographic protocols defined in the Dolev-Yao model. Admitting that cryptographic primitives are idealized, ProVerif checks the security properties of secret, authentication, and observational equivalences. This tool checks the protocol for an unlimited number of executions (sessions). The protocols are modeled and checked using the process calculation syntax of Blanchet et al. [48].

As presented in Figure 4, we have defined channels for data transmission among the network entities. The first channel, ch1, is used to transmit requests and responses between the user and the GWN, and the second channel ch2 is used to establish communication between the GWN and the IoT device. Base types, constants, variables are also defined in this part. Furthermore, hash function, XOR and ECC operations and some auxiliary procedures are also presented. Figure 5 shows the events and the attacker's query model. Our protocol contains six events, namely ULoginPhase () means the user login phase, UAuthenticationPhase () means user authentication by the gateway, GWNAuthentication () means gateway authentication by the sensor, SNSessionKey () means the calculation of the session key by the IoT device, UserSessionKey () means its calculation by the costumer and SNSAuthenticationPhase () means sensor authentication by the user.

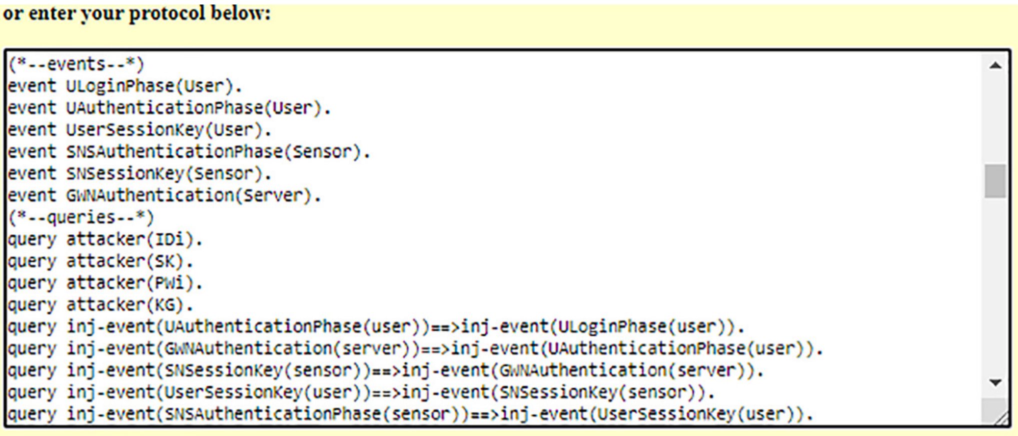
Actions of every entity are structured as in Figures 6, 7, and 8, which show the user process, the gateway process, and the sensor process, respectively. Generally, the user process includes connection request construction, authenticating the GWN and the sensor, and computing the session key. The GWN includes a connection request check, computes the sensor node requisition, verifies the device response, and calculates the parameters of the user response message. The sensor process includes GWN authentication and the response construction. By the end, the three entities are authenticated by each other, and a session key is generated. Figure 9 illustrates the principal process.

Figure 10 illustrates the simulation result. It shows that the mutual authentication process is executed in sequence. In addition, the provided mechanism may ensure the security of the session key, the user's identity, the user's password, and the GWN private key.



```
(*--the two public channel--*)
free chn1:channel.
free chn2:channel.
(*--the basic type--*)
type User.
type Server.
type Sensor.
type Key.
type nonce.
type fingerprint.
type timestamp.
(*--the basic variables--*)
free IDi: bitstring[private].
free SIDj: bitstring[private].
free BIOi: fingerprint[private].
free KG: bitstring[private].
free SK: bitstring[private].
free PWi: bitstring[private].
free P: bitstring.
free user: User.
free server: Server.
free sensor: Sensor.
fun Hash(bitstring):bitstring.
fun BH(bitstring,fingerprint):bitstring.
fun Gen(fingerprint):bitstring.
fun Rep(fingerprint,bitstring):bitstring.
fun bit_timestamp(timestamp):bitstring.
fun bit_Key(Key):bitstring.
fun bit_nonce(nonce):bitstring.
fun Key_bit(bitstring):Key.
fun EccMul(bitstring,bitstring):bitstring.
fun EccAdd(bitstring,bitstring):bitstring.
reduc forall p:bitstring, m1:bitstring, d:Key, m2:bitstring;
EccSub(EccAdd(p,m1),d,m2)=p.
fun XOR(bitstring,bitstring):bitstring.
equation forall x:bitstring, y:bitstring;
XOR(XOR(x,y),y)=x.
fun Con(bitstring,bitstring):bitstring.
reduc forall x:bitstring, y:bitstring;
Split(Con(x,y))=(x,y).
fun checktimestampfresh(bitstring,bool):bool
reduc forall T:bitstring;
checktimestampfresh(T,true)=true
otherwise forall T:bitstring;
checktimestampfresh(T,false)=false.
```

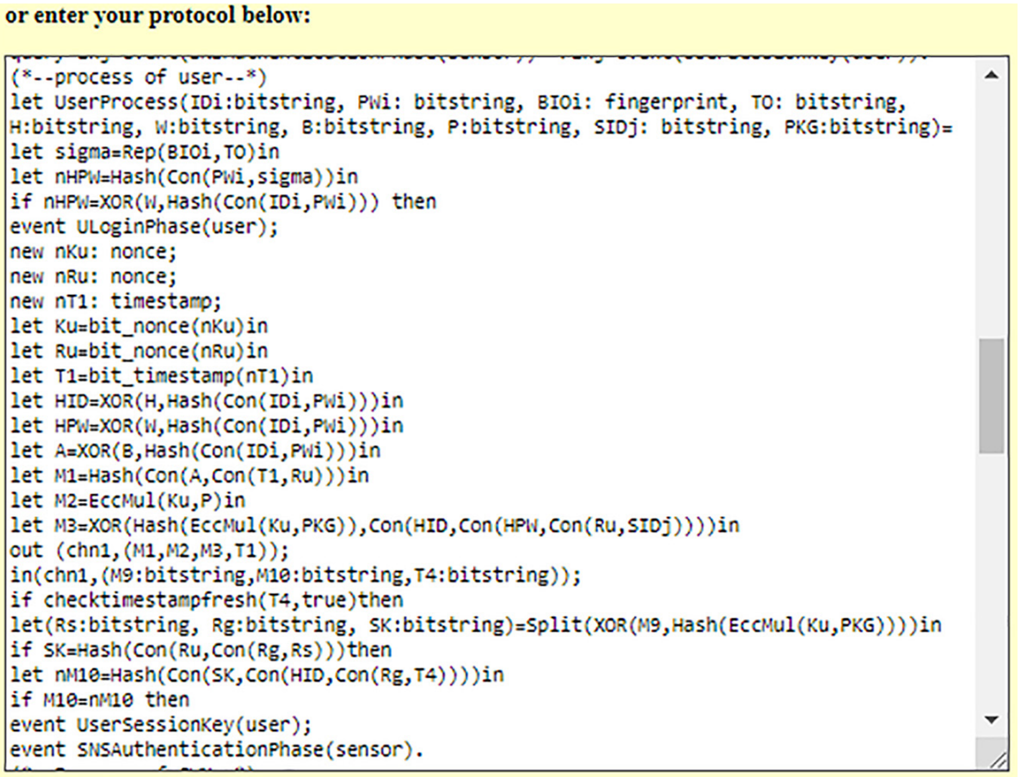
Fig. 4. Definitions



or enter your protocol below:

```
(*--events--*)
event ULoginPhase(User).
event UAuthenticationPhase(User).
event UserSessionKey(User).
event SNSAuthenticationPhase(Sensor).
event SNSessionKey(Sensor).
event GWNAuthentication(Server).
(*--queries--*)
query attacker(IDi).
query attacker(SK).
query attacker(PWi).
query attacker(KG).
query inj-event(UAuthenticationPhase(user))==>inj-event(ULoginPhase(user)).
query inj-event(GWNAuthentication(server))==>inj-event(UAuthenticationPhase(user)).
query inj-event(SNSessionKey(sensor))==>inj-event(GWNAuthentication(server)).
query inj-event(UserSessionKey(user))==>inj-event(SNSessionKey(sensor)).
query inj-event(SNSAuthenticationPhase(sensor))==>inj-event(UserSessionKey(user)).
```

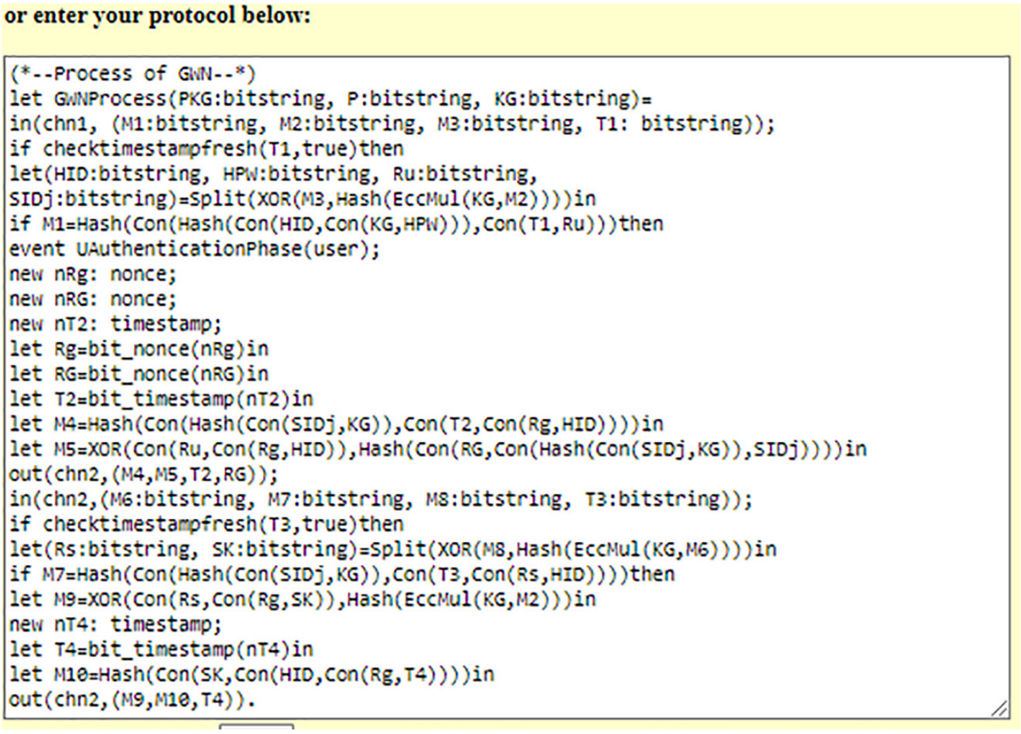
Fig. 5. Events and queries



or enter your protocol below:

```
(*--process of user--*)
let UserProcess(IDi:bitstring, PWi: bitstring, BIOi: fingerprint, TO: bitstring,
H:bitstring, W:bitstring, B:bitstring, P:bitstring, SIDj: bitstring, PKG:bitstring)=
let sigma=Rep(BIOi,TO)in
let nHPW=Hash(Con(PWi,sigma))in
if nHPW=XOR(W,Hash(Con(IDi,PWi))) then
event ULoginPhase(user);
new nKu: nonce;
new nRu: nonce;
new nT1: timestamp;
let Ku=bit_nonce(nKu)in
let Ru=bit_nonce(nRu)in
let T1=bit_timestamp(nT1)in
let HID=XOR(H,Hash(Con(IDi,PWi)))in
let HPW=XOR(W,Hash(Con(IDi,PWi)))in
let A=XOR(B,Hash(Con(IDi,PWi)))in
let M1=Hash(Con(A,Con(T1,Ru)))in
let M2=EccMul(Ku,P)in
let M3=XOR(Hash(EccMul(Ku,PKG)),Con(HID,Con(HPW,Con(Ru,SIDj))))in
out (chn1,(M1,M2,M3,T1));
in(chn1,(M9:bitstring,M10:bitstring,T4:bitstring));
if checktimestampfresh(T4,true)then
let(Rs:bitstring, Rg:bitstring, SK:bitstring)=Split(XOR(M9,Hash(EccMul(Ku,PKG))))in
if SK=Hash(Con(Ru,Con(Rg,Rs)))then
let nM10=Hash(Con(SK,Con(HID,Con(Rg,T4))))in
if M10=nM10 then
event UserSessionKey(user);
event SNSAuthenticationPhase(sensor).
```

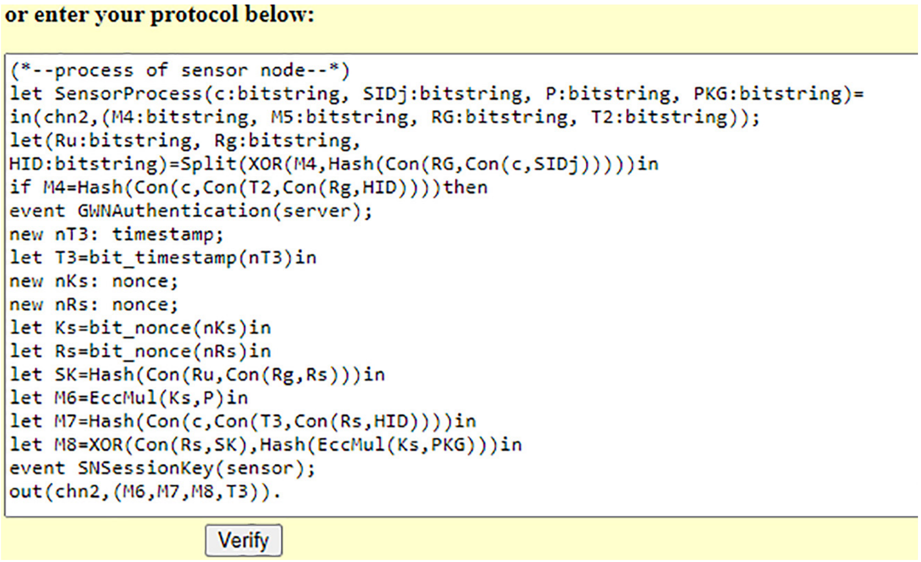
Fig. 6. User process



or enter your protocol below:

```
(*--Process of GWN--*)
let GWNProcess(PKG:bitstring, P:bitstring, KG:bitstring)=
in(chn1, (M1:bitstring, M2:bitstring, M3:bitstring, T1: bitstring));
if checktimestampfresh(T1,true)then
let(HID:bitstring, HPW:bitstring, Ru:bitstring,
SIDj:bitstring)=Split(XOR(M3,Hash(EccMul(KG,M2))))in
if M1=Hash(Con(Hash(Con(HID,Con(KG,HPW))),Con(T1,Ru)))then
event UAuthenticationPhase(user);
new nRg: nonce;
new nRG: nonce;
new nT2: timestamp;
let Rg=bit_nonce(nRg)in
let RG=bit_nonce(nRG)in
let T2=bit_timestamp(nT2)in
let M4=Hash(Con(Hash(Con(SIDj,KG)),Con(T2,Con(Rg,HID))))in
let M5=XOR(Con(Ru,Con(Rg,HID)),Hash(Con(RG,Con(Hash(Con(SIDj,KG)),SIDj))))in
out(chn2,(M4,M5,T2,RG));
in(chn2,(M6:bitstring, M7:bitstring, M8:bitstring, T3:bitstring));
if checktimestampfresh(T3,true)then
let(Rs:bitstring, SK:bitstring)=Split(XOR(M8,Hash(EccMul(KG,M6))))in
if M7=Hash(Con(Hash(Con(SIDj,KG)),Con(T3,Con(Rs,HID))))then
let M9=XOR(Con(Rs,Con(Rg,SK)),Hash(EccMul(KG,M2)))in
new nT4: timestamp;
let T4=bit_timestamp(nT4)in
let M10=Hash(Con(SK,Con(HID,Con(Rg,T4))))in
out(chn2,(M9,M10,T4)).
```

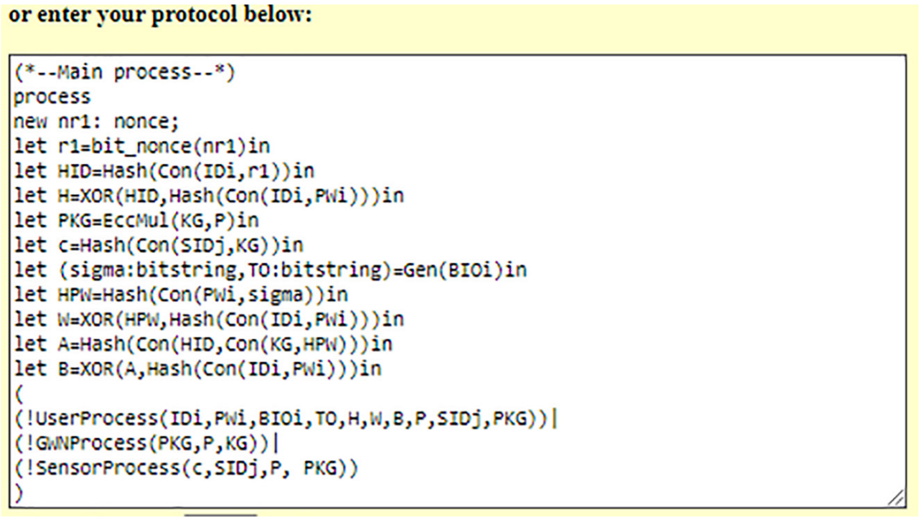
Fig. 7. GWN process



or enter your protocol below:

```
(*--process of sensor node--*)
let SensorProcess(c:bitstring, SIDj:bitstring, P:bitstring, PKG:bitstring)=
in(chn2,(M4:bitstring, M5:bitstring, RG:bitstring, T2:bitstring));
let(Ru:bitstring, Rg:bitstring,
HID:bitstring)=Split(XOR(M4,Hash(Con(RG,Con(c,SIDj)))))in
if M4=Hash(Con(c,Con(T2,Con(Rg,HID))))then
event GWNAuthentication(server);
new nT3: timestamp;
let T3=bit_timestamp(nT3)in
new nKs: nonce;
new nRs: nonce;
let Ks=bit_nonce(nKs)in
let Rs=bit_nonce(nRs)in
let SK=Hash(Con(Ru,Con(Rg,Rs)))in
let M6=EccMul(Ks,P)in
let M7=Hash(Con(c,Con(T3,Con(Rs,HID))))in
let M8=XOR(Con(Rs,SK),Hash(EccMul(Ks,PKG)))in
event SNSessionKey(sensor);
out(chn2,(M6,M7,M8,T3)).
```

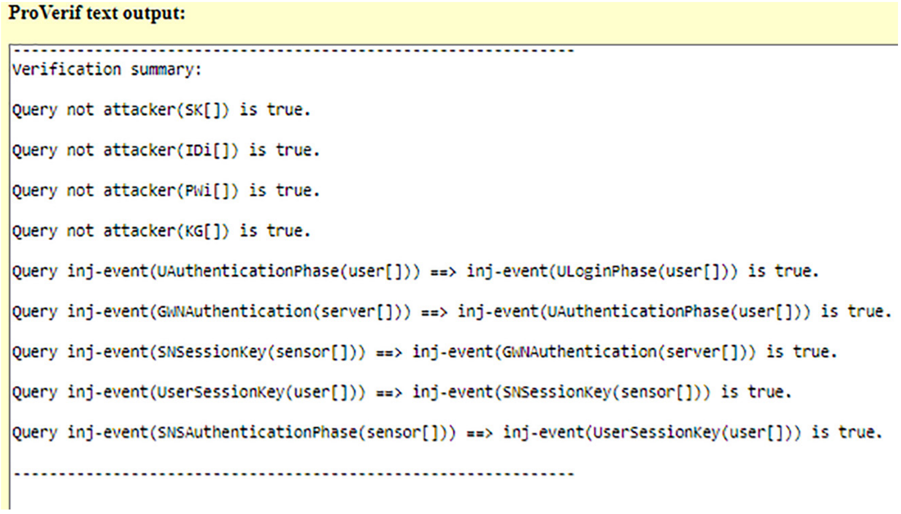
Fig. 8. Sensor process



or enter your protocol below:

```
(*--Main process--*)
process
new nr1: nonce;
let r1=bit_nonce(nr1)in
let HID=Hash(Con(IDi,r1))in
let H=XOR(HID,Hash(Con(IDi,PWi)))in
let PKG=EccMul(KG,P)in
let c=Hash(Con(SIDj,KG))in
let (sigma:bitstring,TO:bitstring)=Gen(BIOi)in
let HPW=Hash(Con(PWi,sigma))in
let W=XOR(HPW,Hash(Con(IDi,PWi)))in
let A=Hash(Con(HID,Con(KG,HPW)))in
let B=XOR(A,Hash(Con(IDi,PWi)))in
(
(!UserProcess(IDi,PWi,BIOi,TO,H,W,B,P,SIDj,PKG))|
(!GWNProcess(PKG,P,KG))|
(!SensorProcess(c,SIDj,P, PKG))
)
```

Fig. 9. Main process



ProVerif text output:

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

Query not attacker(SK[]) is true.

Query not attacker(IDi[]) is true.

Query not attacker(PWi[]) is true.

Query not attacker(KG[]) is true.

Query inj-event(UAuthenticationPhase(user[])) ==> inj-event(ULoginPhase(user[])) is true.

Query inj-event(GWNAuthentication(server[])) ==> inj-event(UAuthenticationPhase(user[])) is true.

Query inj-event(SNSessionKey(sensor[])) ==> inj-event(GWNAuthentication(server[])) is true.

Query inj-event(UserSessionKey(user[])) ==> inj-event(SNSessionKey(sensor[])) is true.

Query inj-event(SNSAuthenticationPhase(sensor[])) ==> inj-event(UserSessionKey(user[])) is true.

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

Fig. 10. Results

6 DISCUSSION: PERFORMANCE AND COMPARATIVE ANALYSES

In this part, we will present the comparison results of our provided protocol with some other schemes regarding computational requirements, communication needs, and security performance.

6.1 Security performance

Table 2. Security features and resistance against attacks

Protocol	F_1	F_2	F_3	F_4	F_5	F_6	A_1	A_2	A_3	A_4	A_5	A_6	A_7	A_8	A_9	A_{10}	A_{11}
[49]	✓	✓	✓	✓	✓	✓	✓	✓	✓	–	–	–	–	✓	✓	–	–
[50]	✓	–	–	✓	–	✓	–	✓	–	✓	✓	✓	✓	✓	–	–	–
[51]	✓	✓	✓	✓	✓	–	✓	✓	✓	✓	✓	✓	–	✓	✓	–	✓
[52]	✓	✓	✓	✓	✓	–	✓	✓	✓	–	✓	✓	–	✓	✓	–	✓
[53]	✓	✓	–	✓	–	–	✓	✓	–	×	–	✓	–	–	–	–	✓
[54]	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	–	✓	–	✓	–	–	✓
[55]	✓	✓	✓	✓	✓	–	–	✓	–	–	–	–	–	–	✓	–	–
Our protocol	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓

Notes: F_1: mutual authentication, F_2: Anonymity, F_3: unlink ability, F_4: key agreement, F_5: key secrecy, F_6: perfect forward secrecy, A_1: Impersonation attack, A_2: reply attack, A_3: node capture, A_4: DoS attack, A_5: Insider attack, A_6: Stolen verifier, A_7: Denning-SSACO attack, A_8: password guessing, A_9: smart card loss, A_{10}: GWN bypassing, A_{11}: man in the middle.

As mentioned in Table 2 and the security analysis section, the suggested mechanism provides the best achievement, resisting all known IoT raids and delivering all security characteristics required in an authentication scheme, including perfect forward secrecy and mutual authentication. However, the authors of [49] cannot prove that their scheme is safe against stolen verifier, DoS assault, Insider attack, denning-SSACO assault, GWN bypassing, or MITM assault. Authors of [50] cannot ensure that their scheme provides anonymity, unlink ability, key secrecy, or resistance against impersonation attacks, node capture, smart card loss, GWN bypassing, or man in the middle. Authors of [53] cannot prove its resistance against node capture, insider attack, denning-SSACO attack, password guessing, smart card loss attack, GWN bypassing, or that it ensures unlink ability, key secrecy, or forward secrecy. More than that, [53] is not safe regarding DoS attacks. Authors of [55] cannot demonstrate that their protocol provides perfect forward secrecy or withstands impersonation raids, node capture raids, DoS raids, insider raids, stolen verifier, denning-SSACO attacks, password guessing, or GWN bypassing.

6.2 Computation overheads

In this subsection, we will compare the calculation requirement of the login and authentication phase of our scheme with the requirement of some related schemes based on ECC. The result of our comparison is summarized in Table 3.

The notation Th is described as the temporal need for a one-way hashing operation, Te is the temporal need for ECC multiplication, Tsig is the temporal need for a HECDSA signature generation and verification execution, Ts is the temporal need for symmetrical encoding and decoding, and Tf is the temporal need of the fuzzy extractor. The cost of calculating the operation or exclusive is generally overlooked because it requires minimal calculations. According to [23], using a processor: Intel(R) Core(TM) i7-6700@ 3.4 GHz, the medium times of used procedures are as follows: Th ≈ 0.0001, Te ≈ 0.442, Tsig ≈ 3.1920, Ts ≈ 0.0026, and Tf ≈ 0.442.

In our proposed protocol, the user requires 7Th + Tf + 2Te to construct the connection demand, perform the necessary checks, and calculate the session key. The GWN requires 9Th + 2Te to check the connection asking, compute the sensor node requisition, verify the device response, and calculate the parameters of the user response message. The sensor requires 6Th + 2Te to confirm the validity of the verification equations and calculate the response for the gateway. Therefore, for completing the login and authentication phase, our protocol needs 22Th + Tf + 6Te. Overall, all compared protocols have very close requirements, excluding [53]. Although our scheme is not the fastest, it requires only about 3 ms to execute.

Table 3. Computational cost comparison

Scheme	User	Gateway	Sensor	Total	Execution Time (Ms)
[49]	7Th + 3Te	10Th + Te	6Th + 2Te	23Th + 6Te	2,6543
[50]	5Th	6Th + 4Te	2Th + 2Te	13Th + 6Te	2,6533
[51]	7Th + 3Te + 1Tf	7Th + Te	4Th + 2Te	18Th + 6Te + Tf	3,0958
[52]	9Th + 3Te	9Th + Te	7Th + 2Te	25Th + 6Te	2,6545
[53]	–	–	–	15Th + 2Tf + 4Ts + 2Tsig + 6Te	9,9319
[54]	5Th + 3Te	5Th + 2Te + Ts	3Th + 3Te + Ts	13Th + 8Te + 2Ts	3,5425
[55]	–	–	–	Ts + 15Th + 6Te	2,6561
Our protocol	7Th + Tf + 2Te	9Th + 2Te	6Th + 2Te	22Th + Tf + 6Te	3,0962

6.3 Communication overheads

Table 4 presents the comparative results among the suggested scheme and some other schemes regarding communication overheads; it also provides the number of messages communicated in the login and authentication phase. In line with [52], the size of the identity, random nonce, timestamp, hashing function output, symmetrical encoding and decoding block, and the length of a point in an elliptical curve are respectively 128 bits, 128 bits, 32 bits, 160 bits, 256 bits, and 320 bits. The analysis of Tables 2, 3, and 4 shows that the schemes with less computational and communication cost, such as [55], cannot ensure an acceptable resistance against attacks. However, the proposed scheme provides an acceptable communication cost compared to related schemes, given their resistance against attacks and computational costs.

Table 4. Communication cost comparison

Protocol	Total Messages	Total Communication Cost
[49]	4	3456
[50]	4	2304
[51]	3	2112
[52]	4	3552
[53]	3	–
[54]	6	3456
[55]	4	2144
Our protocol	4	2912

Based on this performance analysis, the proposed scheme required tolerable communication and computation costs while it provided a high level of security requirements and resisted all well-known attacks as mentioned before. Therefore, we may conclude that the combination of ECC with user biometrics presents a multi-factor authentication scheme that outperforms other existing methods, particularly in real-world scenarios, considering that its execution time is only 3 ms, which would show no appreciable retardation in the eyes of any human being.

7 CONCLUSION

Overall, IoT security has been a real issue in its deployment, the thing that leads researchers to design a diversity of authentication schemes to secure exchanged data in IoT networks. However, most of those schemes present some vulnerabilities, especially in guaranteeing the anonymity of the customer. As far as we know, the most part of IoT authentication schemes are built on two factors. In our paper, we provided an anonymous authentication scheme built on three factors using ECC, which combine knowledge, possession, and attributes to provide a high level of safety. Then, the results of our informal analyses proved that the protocol is robust regarding well-known IoT assaults, particularly the Denning-SSACO attack, smart card loss, GWN bypassing, and MITM. Thereafter, the formal analyses under the ProVerif simulator confirm that our protocol answers all security requirements. Ultimately, we compared our protocol with some related schemes based also on ECC. This comparison demonstrates that the proposed mechanism presents a reduced cost of computation and communication regarding its security level.

8 REFERENCES

[1] S. Dargaoui, M. Azrour, A. El Allaoui, A. Guezzaz, A. Alabdulatif, and A. Alnajim, "An exhaustive survey on authentication classes in the IoT environments," *Indonesian Journal of Electrical Engineering and Informatics (IJEEI)*, vol. 12, no. 1, pp. 15–31, 2024. https://doi.org/10.52549/ijeei.v12i1.5170

[2] S. Dargaoui, M. Azrour, A. El allaoui, A. Guezzaz, A. Alabdulatif, and A. Alnajim, "Internet of things authentication protocols: Comparative study," *Computers, Materials & Continua*, vol. 79, no. 1, pp. 65–91, 2024. https://doi.org/10.32604/cmc.2024.047625

[3] S. Dargaoui *et al.*, "An overview of the security challenges in IoT environment," in *Advanced Technology for Smart Environment and Energy*, J. Mabrouki, A. Irshad, and A. Choudhry, Eds., Springer, Cham, 2023, pp. 151–160. https://doi.org/10.1007/978-3-031-25662-2_13

[4] S. Dargaoui *et al.*, "Security issues in internet of medical things," in *Blockchain and Machine Learning for IoT Security*, 2023, pp. 77–91. https://doi.org/10.1201/9781003438779-5

[5] M. Azrour *et al.*, "A survey of machine and deep learning applications in the assessment of water quality," in *Technical and Technological Solutions Towards a Sustainable Society and Circular Economy*, *World Sustainability Series*, J. Mabrouki and A. Mourade, Eds., Springer, Cham, 2024, pp. 471–483. https://doi.org/10.1007/978-3-031-56292-1_38

[6] S. Dargaoui *et al.*, "Applications of blockchain in healthcare: Review study," in *IoT, Machine Learning and Data Analytics for Smart Healthcare*, 2024, pp. 1–12. https://doi.org/10.1201/9781003430735-1

[7] S. Dargaoui, M. Azrour, A. El Allaoui, A. Guezzaz, and S. Benkirane, "Authentication in Internet of Things: State of art," in *Proceedings of the 6th International Conference on Networking, Intelligent Systems & Security (NISS '23), Association for Computing Machinery*, 2023, pp. 1–6. https://doi.org/10.1145/3607720.3607723

[8] C. Hazman, A. Guezzaz, S. Benkirane, and M. Azrour, "Enhanced IDS with deep learning for iot-based smart cities security," *Tsinghua Science and Technology*, vol. 29, no. 4, pp. 929–947, 2024. https://doi.org/10.26599/TST.2023.9010033

[9] A. E. M. Eljialy, M. Y. Uddin, and S. Ahmad, "Novel framework for an intrusion detection system using multiple feature selection methods based on deep learning," *Tsinghua Science and Technology*, vol. 29, no. 4, pp. 948–958, 2024. https://doi.org/10.26599/TST.2023.9010032

[10] M. Azrour, J. Mabrouki, A. Guezzaz, S. Benkirane, and H. Asri, "Implementation of realtime water quality monitoring based on Java and Internet of Things," in *Integrating Blockchain and Artificial Intelligence for Industry 4.0 Innovations, EAI/Springer Innovations in Communication and Computing*, S. Goundar and R. Anandan, Eds., Springer, Cham, 2023, pp. 133–143. https://doi.org/10.1007/978-3-031-35751-0_8

[11] S. Dargaoui *et al.*, "Internet-of-Things-Enabled Smart Agriculture: Security Enhancement Approaches," in *2024 4th International Conference on Innovative Research in Applied Science, Engineering and Technology (IRASET)*, 2024, pp. 1–5. https://doi.org/10.1109/IRASET60544.2024.10548705

[12] A. Balasundaram, S. Routray, A. V. Prabu, P. Krishnan, P. P. Malla, and M. Maiti, "Internet of things (IoT) based smart healthcare system for efficient diagnostics of health parameters of patients in emergency care," *IEEE Internet of Things Journal*, vol. 10, no. 21, pp. 18563–18570, 2023. https://doi.org/10.1109/JIOT.2023.3246065

[13] A. Aditya, S. Anwarul, R. Tanwar, and S. K. V. Koneru, "An IoT assisted intelligent parking system (IPS) for smart cities," *Procedia Computer Science*, vol. 218, pp. 1045–1054, 2023. https://doi.org/10.1016/j.procs.2023.01.084

[14] C.-T. Chen, C.-C. Lee, and I.-C. Lin, "Correction: Efficient and secure three-party mutual authentication key agreement protocol for WSNs in IoT environments," *PLoS ONE*, vol. 15, no. 6, p. e0234631, 2020. https://doi.org/10.1371/journal.pone.0234631

[15] D. Kumar, S. Chand, and B. Kumar, "Cryptanalysis and improvement of a user authentication scheme for wireless sensor networks using chaotic maps," *IET Networks*, vol. 9, no. 6, pp. 315–325, 2020. https://doi.org/10.1049/iet-net.2019.0009

[16] D. Kaur, D. Kumar, K. K. Saini, and H. S. Grover, "An improved user authentication protocol for wireless sensor networks," *Transactions on Emerging Telecommunications Technologies*, vol. 30, no. 10, p. e3745, 2019. https://doi.org/10.1002/ett.3745

[17] J. Oh, S. Yu, J. Lee, S. Son, M. Kim, and Y. Park, "A secure and lightweight authentication protocol for IoT-based smart homes," *Sensors*, vol. 21, no. 4, p. 1488, 2021. https://doi.org/10.3390/s21041488

[18] P. K. Panda and S. Chattopadhyay, "A secure mutual authentication protocol for IoT environment," *Journal of Reliable Intelligent Environments*, vol. 6, pp. 79–94, 2020. https://doi.org/10.1007/s40860-020-00098-y

[19] M. Dammak, O. R. M. Boudia, M. A. Messous, S. M. Senouci, and C. Gransart, "Token-based lightweight authentication to secure IoT networks," in *2019 16th IEEE Annual Consumer Communications & Networking Conference (CCNC)*, 2019, pp. 1–4. https://doi.org/10.1109/CCNC.2019.8651825

[20] L. Kou, Y. Shi, L. Zhang, D. Liu, and Q. Yang, "A lightweight three-factor user authentication protocol for the information perception of IoT," *CMC-Computers, Materials & Continua*, vol. 58, no. 2, pp. 545–565, 2019. https://doi.org/10.32604/cmc.2019.03760

[21] T. M. Butt, R. Riaz, C. Chakraborty, S. S. Rizvi, and A. Paul, "Cogent and energy efficient authentication protocol for WSN in IoT," *Comput. Mater. Contin.*, vol. 68, no. 2, pp. 1877–1898, 2021. https://doi.org/10.32604/cmc.2021.014966

[22] V. O. Nyangaresi, "Lightweight anonymous authentication protocol for resource-constrained smart home devices based on elliptic curve cryptography," *Journal of Systems Architecture*, vol. 133, p. 102763, 2022. https://doi.org/10.1016/j.sysarc.2022.102763

[23] J. Cui, F. Cheng, H. Zhong, Q. Zhang, C. Gu, and L. Liu, "Multi-factor based session secret key agreement for the industrial Internet of Things," *Ad Hoc Networks*, vol. 138, p. 102997, 2023. https://doi.org/10.1016/j.adhoc.2022.102997

[24] S. Yu and K. Park, "ISG-SLAS: Secure and lightweight authentication and key agreement scheme for industrial smart grid using fuzzy extractor," *Journal of Systems Architecture*, vol. 131, p. 102698, 2022. https://doi.org/10.1016/j.sysarc.2022.102698

[25] R. Krishnasrija, A. K. Mandal, and A. Cortesi, "A lightweight mutual and transitive authentication mechanism for IoT network," *Ad Hoc Networks*, vol. 138, p. 103003, 2023. https://doi.org/10.1016/j.adhoc.2022.103003

[26] J. Lee *et al.*, "PUFTAP-IoT: PUF-based three-factor authentication protocol in IoT environment focused on sensing devices," *Sensors*, vol. 22, no. 18, p. 7075, 2022. https://doi.org/10.3390/s22187075

[27] A. K. Yadav, M. Misra, P. K. Pandey, and M. Liyanage, "An EAP-based mutual authentication protocol for WLAN connected IoT devices," *IEEE Transactions on Industrial Informatics*, vol. 19, no. 2, pp. 1343–1355, 2023. https://doi.org/10.1109/TII.2022.3194956

[28] P. Bagga, A. Mitra, A. K. Das, P. Vijayakumar, Y. Park, and M. Karuppiah, "Secure biometric-based access control scheme for future IoT-enabled cloud-assisted video surveillance system," *Computer Communications*, vol. 195, pp. 27–39, 2022. https://doi.org/10.1016/j.comcom.2022.08.003

[29] N. Garg, R. Petwal, M. Wazid, D. P. Singh, A. K. Das, and J. J. Rodrigues, "On the design of an AI-driven secure communication scheme for internet of medical things environment," *Digital Communications and Networks*, vol. 9, no. 5, pp. 1080–1089, 2022. https://doi.org/10.1016/j.dcan.2022.04.009

[30] S. K. Dwivedi, R. Amin, and S. Vollala, "Design of secured blockchain based decentralized authentication protocol for sensor networks with auditing and accountability," *Computer Communications*, vol. 197, pp. 124–140, 2023. https://doi.org/10.1016/j.comcom.2022.10.016

[31] S. Rostampour, N. Bagheri, Y. Bendavid, M. Safkhani, S. Kumari, and J. J. P. C. Rodrigues, "An authentication protocol for next generation of constrained Iot systems," *IEEE Internet of Things Journal*, vol. 9, no. 21, pp. 21493–21504, 2022. https://doi.org/10.1109/JIOT.2022.3184293

[32] R. Kumar, S. Singh, and P. K. Singh, "A secure and efficient computation based multifactor authentication scheme for Intelligent IoT-enabled WSNs," *Computers and Electrical Engineering*, vol. 105, p. 108495, 2023. https://doi.org/10.1016/j.compeleceng.2022.108495

[33] B. Khalid, K. N. Qureshi, K. Z. Ghafoor, and G. Jeon, “An improved biometric based user authentication and key agreement scheme for intelligent sensor based wireless communication,” *Microprocessors and Microsystems*, vol. 96, p. 104722, 2023. https://doi.org/10.1016/j.micpro.2022.104722

[34] R. Hajian, A. Haghighat, and S. H. Erfani, “A secure anonymous D2D mutual authentication and key agreement protocol for IoT,” *Internet of Things*, vol. 18, p. 100493, 2022. https://doi.org/10.1016/j.iot.2021.100493

[35] C. Patel, A. K. Bashir, A. A. AlZubi, and R. H. Jhaveri, “EBAKE-SE: A novel ECC-based authenticated key exchange between industrial IoT devices using secure element,” *Digital Communications and Networks*, vol. 9, no. 2, pp. 358–366, 2022. https://doi.org/10.1016/j.dcan.2022.11.001

[36] M. Azrour, J. Mabrouki, and R. Chaganti, “New efficient and secured authentication protocol for remote healthcare systems in cloud-IoT,” *Security and Communication Networks*, vol. 2021, no. 1, p. 5546334, 2021. https://doi.org/10.1155/2021/5546334

[37] S. Dargaoui *et al.*, “IoT-driven smart agriculture: Security issues and authentication schemes classification,” in *Proceeding of the International Conference on Connected Objects and Artificial Intelligence (COCIA2024)*, in Lecture Notes in Networks and Systems, Y. Mejdoub and A. Elamri, Eds., Springer, Cham, 2024, vol. 1123, pp. 61–66. https://doi.org/10.1007/978-3-031-70411-6_10

[38] P. Gope and T. Hwang, “A realistic lightweight anonymous authentication protocol for securing real-time application data access in wireless sensor networks,” *IEEE Transactions on Industrial Electronics*, vol. 63, no. 11, pp. 7124–7132, 2016. https://doi.org/10.1109/TIE.2016.2585081

[39] A. Ghani, K. Mansoor, S. Mehmood, S. A. Chaudhry, A. U. Rahman, and M. Najmus Saqib, “Security and key management in IoT-based wireless sensor networks: An authentication protocol using symmetric key,” *International Journal of Communication Systems*, vol. 32, no. 16, p. e4139, 2019. https://doi.org/10.1002/dac.4139

[40] D. Q. Bala, S. Maity, and S. K. Jena, “Mutual authentication for IoT smart environment using certificate-less public key cryptography,” in *2017 Third International Conference on Sensing, Signal Processing and Security (ICSSS)*, 2017, pp. 29–34. https://doi.org/10.1109/SSPS.2017.8071559

[41] N. Li, D. Liu, and S. Nepal, “Lightweight mutual authentication for IoT and its applications,” *IEEE Transactions on Sustainable Computing*, vol. 2, no. 4, pp. 359–370, 2017. https://doi.org/10.1109/TSUSC.2017.2716953

[42] Q. Jiang, J. Ma, F. Wei, Y. Tian, J. Shen, and Y. Yang, “An untraceable temporal-credential-based two-factor authentication scheme using ECC for wireless sensor networks,” *Journal of Network and Computer Applications*, vol. 76, pp. 37–48, 2016. https://doi.org/10.1016/j.jnca.2016.10.001

[43] D. He, N. Kumar, and N. Chilamkurti, “A secure temporal-credential-based mutual authentication and key agreement scheme with pseudo identity for wireless sensor networks,” *Information Sciences*, vol. 321, pp. 263–277, 2015. https://doi.org/10.1016/j.ins.2015.02.010

[44] X. Li, J. Niu, S. Kumari, F. Wu, A. K. Sangaiah, and K.-K. R. Choo, “A three-factor anonymous authentication scheme for wireless sensor networks in Internet of Things environments,” *Journal of Network and Computer Applications*, vol. 103, pp. 194–204, 2018. https://doi.org/10.1016/j.jnca.2017.07.001

[45] M. A. Qureshi and A. Munir, “PUF-IPA: A PUF-based identity preserving protocol for Internet of Things authentication,” in *2020 IEEE 17th Annual Consumer Communications & Networking Conference (CCNC)*, 2020, pp. 1–7. https://doi.org/10.1109/CCNC46108.2020.9045264

[46] M. T. Hammi, B. Hammi, P. Bellot, and A. Serhrouchni, "Bubbles of trust: A decentralized blockchain-based authentication system for IoT," *Computers & Security*, vol. 78, pp. 126–142, 2018. https://doi.org/10.1016/j.cose.2018.06.004

[47] D. Dolev and A. Yao, "On the security of public key protocols," *IEEE Transactions on Information Theory*, vol. 29, no. 2, pp. 198–208, 1983. https://doi.org/10.1109/TIT.1983.1056650

[48] B. Blanchet, B. Smyth, V. Cheval, and M. Sylvestre, "Automatic cryptographic protocol verifier, user manual and tutorial," CNRS, Departement d'Informatique, Ecole Normale Superieure, Paris, 2015.

[49] B. Hu, W. Tang, and Q. Xie, "A two-factor security authentication scheme for wireless sensor networks in IoT environments," *Neurocomputing*, vol. 500, pp. 741–749, 2022. https://doi.org/10.1016/j.neucom.2022.05.099

[50] M. Azrour, J. Mabrouki, A. Guezzaz, and Y. Farhaoui, "New enhanced authentication protocol for Internet of Things," *Big Data Mining and Analytics*, vol. 4, no. 1, pp. 1–9, 2021. https://doi.org/10.26599/BDMA.2020.9020010

[51] Q. Xie, Z. Ding, and B. Hu, "A secure and privacy-preserving three-factor anonymous authentication scheme for wireless sensor networks in Internet of Things," *Security and Communication Networks*, vol. 2021, no. 1, pp. 1–12, 2021. https://doi.org/10.1155/2021/4799223

[52] X. Wang, Y. Teng, Y. Chi, and H. Hu, "A robust and anonymous three-factor authentication scheme based ECC for smart home environments," *Symmetry*, vol. 14, no. 11, p. 2394, 2022. https://doi.org/10.3390/sym14112394

[53] J. Pirayesh, A. Giaretta, M. Conti, and P. Keshavarzi, "A PLS-HECC-based device authentication and key agreement scheme for smart home networks," *Computer Networks*, vol. 216, p. 109077, 2022. https://doi.org/10.1016/j.comnet.2022.109077

[54] M. A. Khan, B. A. Alzahrani, A. Barnawi, A. Al-Barakati, A. Irshad, and S. A. Chaudhry, "A resource friendly authentication scheme for space–air–ground–sea integrated Maritime Communication Network," *Ocean Engineering*, vol. 250, p. 110894, 2022. https://doi.org/10.1016/j.oceaneng.2022.110894

[55] A. K. Yadav, M. Misra, P. K. Pandey, K. Kaur, S. Garg, and X. Chen, "A Provably Secure ECC-based Multi-factor 5G-AKA Authentication Protocol," in *GLOBECOM 2022–2022 IEEE Global Communications Conference*, IEEE, 2022, pp. 516–521.

9 AUTHORS

Souhayla Dargaoui is a PhD candidate at the Engineering Science and Technology Laboratory, IDMS Team, Faculty of Sciences and Techniques, Moulay Ismail University of Meknes, Morocco. She received M.Eng. in Computer Science and Complex Systems Engineering from the same faculty in 2022, and M.Eng. in Communication Systems and embedded Electronics from the National school of Applied Sciences, Abdelmalek Assaadi University, Tangier, Morocco in 2020. Her research interests include authentication protocols, smart systems security, and Internet of Things (E-mail: s.dargaoui@edu.umi.ac.ma).

Mourade Azrour received his PhD from Faculty of Sciences and Techniques, Moulay Ismail University of Meknes, Morocco. He has received his MS in Computer and Distributed Systems from the Faculty of Sciences, Ibn Zohr University, Agadir, Morocco in 2014. Mourade currently works as computer sciences professor at the Department of Computer Science, Faculty of Sciences and Techniques, Moulay Ismail University of Meknes. His study interests include Authentication protocol, Computer Security, Internet of things, Smart systems, Machine learning and so ones.

Mourade is member of the member of the scientific committee of numerous international conferences. He is also a reviewer of various scientific journals. He has published more than 127 scientific papers and book chapters. Mourade has edited many scientific books too.

Ahmad El Allaoui is an Assistant Professor at the Department of Computer Science, Faculty of Sciences and Techniques, Moulay Ismail University. He is an IDMS Team member. He focuses on semantic image segmentation, medical imaging, classification algorithms, segmentation, image processing, evolutionary algorithms, and genetic algorithms.

Azidine Guezzaz received MS degree in Computer Science and Distributed Systems from the Department of Mathematics and Computer Science, Faculty of Science, University Ibn Zohr, Agadir, Morocco in 2013. He received the PhD degree from Faculty of Science, University Ibn Zohr, Agadir, Morocco in 2018. He was a Professor at the Technology High School and BTS in the period 2014–2018. He then joined Cadi Ayyad University in 2018 as an Assistant Professor. His field of research interests include intrusion detection and prevention, computer and network security, and cryptography.

Abdulatif Alabdulatif is an Assistant Professor at the College of Computer, Qassim University, Saudi Arabia. He received the PhD degree in Computer Science from RMIT University, Australia. He received his M.Sc. degree in Computer Science from the same university. His study interests include applied cryptography, cloud computing, and data mining. His study interests include artificial intelligence, machine learning, information retrieval, VOIP, and wireless networks (E-mail: ab.alabdulatif@qu.edu.sa).

Sultan Ahmad (Member, IEEE) received the master's degree (Hons.) in computer science and applications from the Prestigious Aligarh Muslim University, India. He graduated in Computer Science and Applications in 2002 from Patna University, India. He is currently working as a Lecturer with the Department of Computer Science, College of Computer Engineering and Sciences, Prince Sattam Bin Abdulaziz University, Al-Kharj, Saudi Arabia. He has more than 12 years of teaching and research experience. His study and teaching interests include cloud computing, big data, machine learning, and the Internet of Things. He has presented his study papers in many national and international conferences and published research articles in many peer-reviewed reputed journals. He is a member of IACSIT and Computer Society of India.