Journal of Scientific & Industrial Research
Vol. 85, June 2026, pp. 538-546
DOI: 10.56042/jsir.v85i6.23070

A Lightweight Authentication Protocol for WSNs in Industry 4.0

Khushil Kumar Saini[1], Devender Kumar[2]* & Bijendra Kumar[1]

[1]Department of Computer Science and Engineering, [2]Department of Information Technology,
Netaji Subhas University of Technology, New Delhi 110 078, India

Received 24 August 2025; revised 10 March 2026; accepted 21 August 2026

The advent of Industry 4.0 has significantly accelerated the integration of Wireless Sensor Networks (WSNs) across diverse industrial environments, enabling real-time monitoring, intelligent decision-making and predictive maintenance. These sensor-driven cyber-physical systems form the backbone of smart factories by facilitating seamless data exchange across machines, control systems, and cloud platforms. Moreover, the deployment of WSNs enhances supply chain visibility, asset tracking, and environmental monitoring, contributing to improved safety, efficiency, and sustainability. However, the integration of WSNs into industrial ecosystems exposes them to a variety of security vulnerabilities, like DDoS, device hijacking, man-in-the-middle attacks, and ransomware, which may compromise operational safety and the confidentiality of critical industrial data. Traditional security mechanisms, though effective in general computing environments, are often unsuitable for WSNs due to high processing delays and energy demands. In Industry 4.0, IoT and WSN devices have very limited computation capability, memory, and energy. In such a scenario, light weight authentication schemes enable secure communication with minimal overhead and hence support low-latency applications such as robotic coordination where delays are critical. This paper put forward a lightweight authentication protocol purpose-built for industrial automation scenarios, employing hash function and symmetric key cryptography to ensure mutual authentication and secure data exchange with minimal computational overhead. To confirm that the protocol satisfies its intended security goals, it is evaluated using two methods: automated formal verification with the ProVerif tool complemented by an additional informal security discussion. A comparative evaluation against existing schemes demonstrates that the proposed scheme offers an effective and resilient approach for addressing the evolving security challenges of Industry 4.0.

Keywords: Attacks, Automation, Cryptography, Industrial internet of things, ProVerif

Introduction

The integration of interconnected smart devices with traditional industrial systems, widely known as Industry 4.0, has fundamentally transformed the operation of modern manufacturing and production systems. This builds upon the foundations of previous industrial phases - beginning with mechanization introduced by steam-power (Industry 1.0), progressing to the large-scale production driven by electricity (Industry 2.0), and later evolving into automation made possible through electronics and information technology (Industry 3.0). A defining feature of Industry 4.0 is the focus on smart and interconnected ecosystems, where machines and systems autonomously monitor their states, exchange information, and adapt to changing conditions. This fourth industrial revolution builds upon Cyber-Physical Systems (CPS) that enable seamless interaction between digital and physical processes, Wireless Sensor Networks (WSNs) that provide device connectivity, AI and ML to support predictive operations, Big Data Analytics for actionable insights, and Cloud Computing for scalable data management. Wireless Sensor Networks (WSNs) are a core enabling technology for connected industrial systems, forming the backbone of the Industrial Internet of Things (IIoT). In Industry 4.0, WSNs enable industrial environments to monitor conditions, exchange information, and gather data with minimal disruption. Composed of distributed low-power sensors, WSNs are widely used for monitoring machine health, temperature, vibration, pressure, humidity, and energy consumption, ensuring early fault detection and predictive maintenance in the in industrial automation. They also play a critical role in process optimization, by facilitating adaptive control and reducing downtime in production lines. Moreover, the deployment of WSNs enhances supply chain visibility, asset tracking, and environmental monitoring, contributing to improved safety, efficiency, and sustainability.

*Author for Correspondence
E-mail: devender.kumar@nsut.ac.in

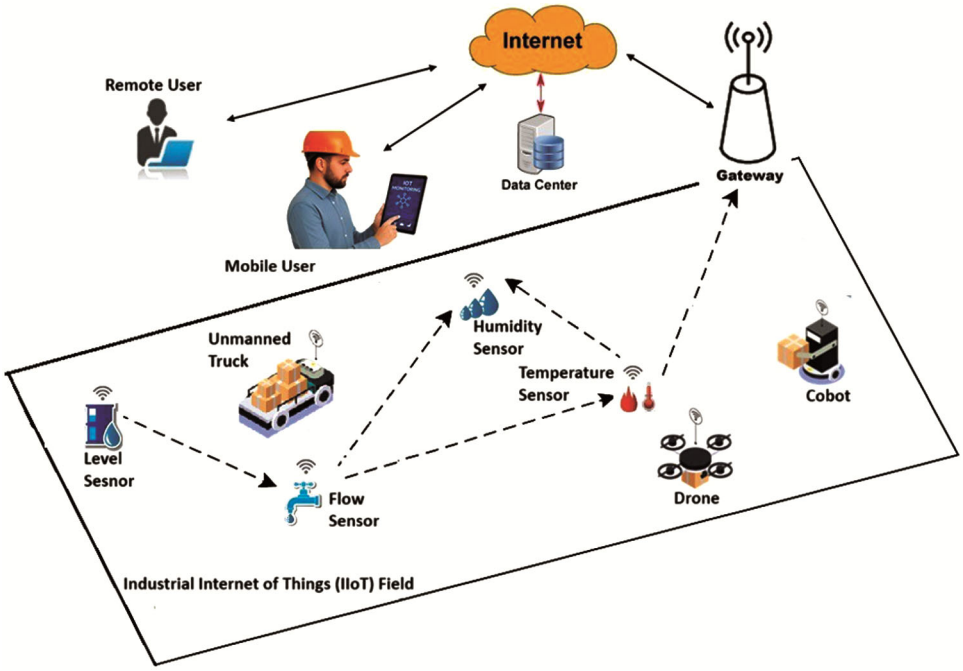


Fig. 1 — Basic architecture of the WSN in IIoT

Basic architecture of a WSN in an IIoT field is shown in Fig. 1. Within the IIoT architecture, three primary communication entities of the WSNs are: i) sensor nodes, that sense and capture the data from the field of deployment, perform processing and organizing operations on them, and then communicate to a nearby gateway; ii) gateway node, which is used to inter-connect two networks with differing transmission protocols and facilitates the seamless data transmission between the wireless devices and the network and; iii) users, who are equipped with a mobile device to access the sensor. The sensor nodes work with highly limited computational abilities and energy resources as these are usually powered by battery, hence WSNs in Industry 4.0 face challenges such as limited computational resources and energy constraints. Highly interconnect nature of devices in IIoT exposes the industries to cyber threats such as data breaches, and attacks on critical infrastructure. Hence, addressing these challenges requires lightweight, secure, and energy-efficient protocols that can ensure reliability and trust in time-sensitive industrial applications.

Literature Review

Das[1] discusses an authentication protocol that employs smart cards to enhance security in WSNs. He *et al.*[2] cryptanalyze the protocol of Das[1] and show that there are several weaknesses, including failure to resist impersonation attacks and threats from insiders having privileged access. To handle these shortcomings, a two-factor authentication mechanism with enhanced security is introduced by He *et al.*[2] Alghathbar & Khan[3] observed that password update mechanism is not present in Das's protocol, leading them to develop their own two-factor authentication scheme. Subsequently, a secure user authentication procedure is introduced by Yoon *et al.*[4], which is later analyzed by He *et al.*[5], who observe the scheme of Yoon *et al.*[4] to be prone to sensor impersonation attacks, prompting them to propose an improved protocol. Kim *et al.*[6] suggest an authentication mechanism making use of two factors; however, Chang *et al.*[7] identify many flaws in it, like vulnerability to a variety of threats such as impersonation, Man-in-The-Middle (MITM) and smart card loss. To address these concerns, an authentication protocol is developed based on dynamic identity by Chang *et al.*[7]

Farash *et al.*[8] design an authentication scheme using hash function, which is later studied by Amin *et al.*[9] and is found to be suffering from attacks like session-specific and offline password guessing. Park *et al.*[10] examined the protocol suggested by Chang *et al.*[7] and identified several weaknesses, including lack of secure password updates and susceptibility to offline password guessing attacks. For overcoming these weaknesses, they suggested a biometric-based

authentication mechanism using Elliptic Curve Cryptography (ECC). Amin *et al.*[11] proposed an authentication protocol, which is cryptanalyzed by Lu *et al.*[12], who showed that it fails to prevent identity leakage, password guessing, and impersonation attacks. Subsequently, Lu *et al.*[12] proposed an authentication scheme specifically designed for WSNs. Lee *et al.*[13] suggested an authentication method utilizing chaotic maps; however, Jabbari & Mohasefi[14] pointed out that it is not able to ensure user anonymity and resist impersonation attacks. They subsequently proposed an improved variant using chaotic maps. To minimize communication and storage overheads, Tewari & Gupta[15] design a lightweight scheme for IoT environments to implement mutual authentication.

Li *et al.*[16] introduced a mutual authentication method utilizing chaotic maps, while a biometrics-based protocol is presented by Das *et al.*[17] Hussain & Chaudhry[18] analysed the scheme of Das *et al.*[17] and revealed that it neither preserve user untraceability nor achieves perfect forward secrecy, and remains susceptible to smart card theft. Similarly, Kaur *et al.*[19] noticed that the protocol of Li *et al.*[16] fails to prevent attacks like an adversary stealing the device, offline guessing of password, malicious actions by insiders, the replay of intercepted messages, and additionally lacks user untraceability. To handle these shortcomings, Kumar *et al.*[20] introduced an enhanced anonymous authentication protocol utilizing Physically Unclonable Functions (PUFs), addressing the problems like impersonation attacks, smart card loss, and stolen verifier observed in an earlier scheme of Gope & Hwang.[21] Kumar[22] introduced a secure, lightweight authentication protocol incorporating a fuzzy verifier mechanism to withstand offline password guessing attacks. However, Mo *et al.*[23] reports that the scheme presented by Kumar[22] is prone to leakage of information because the three random numbers used for session key generation are not independent of one another. Further analysis by Neerul & Kumar[24] reveals additional security flaws, including vulnerability to password guessing attacks, privileged insider threats, session-specific random number leakage, as well as limitation in providing perfect forward secrecy and user untraceability. Authentication in Wireless Sensor Networks remains a very significant challenge. Although, many sophisticated authentication schemes have been designed by researchers in the past as discussed above, however, these protocols suffer from a lot of security flaws restricting their deployment in real-world environments. Given the existence of such vulnerabilities across the existing protocols, there is an undeniable requirement for a better solution. Hence, this study introduces a security-enhanced authentication method for WSNs, making use of symmetric key cryptography, to ensure improved resistance against various known attacks.

Adversary Model

In this work, the assumed threat model provides the following capabilities to attacker A:

- An administrator or insider can be a potential eavesdropper.
- A can physically acquire the smart card and can retrieve all the sensitive data.
- A can have complete access over the channel used for communication, through which messages between the server and user can be intercepted, modified, and replayed.[25]
- Within polynomial time, A can successfully deduce the user's identity and password.[26,27]

Key Contribution

Contribution of the current study is as presented below:

- This study introduced a superior authentication scheme for WSNs employing asymmetric key cryptography.
- Informal security analysis of the proposed scheme confirms its capability to resist a wide array of threats against a broad range of established attacks.
- ProVerif is employed to perform the formal verification of the proposed scheme, confirming its resilience against both active and passive attacks.
- A thorough comparison with the existing protocols of relevant studies[28–30] reveals that the proposed scheme delivers enhanced security features.

Proposed Protocol

The proposed scheme comprises three key phases: 1) Registration, covering both sensor and user, 2) Authentication and Key agreement and 3) Password Update. The symbols used in this study and their corresponding descriptions are presented in Table 1.

Registration Phase

Sensor node registration: Sy chooses its identity SIDy and transmits this identity along with K2 to the gateway node via a private channel. GWN selects its

Table1 — Description of symbols

Symbol	Description
S_y	Sensornode
GWN	Gateway node
U_x	User
ID_x	U_x's Identity
PW_x	U_x's Password
B_x	U_x's biometrics
A	Adversary
K_1	GWN's masterkey
SC	Smartcard
S_{Key}	Sessionkey
SID_y	Sensor identity
K_2	S 's secret value
TS_i	Timestamp
ΔT	Timewindow
En_k	Encryption with symmetric keyk
Dec_k	Decryption with symmetric key k
||	Concatenation operation
⊕	XOR operation
h(.)	Hash function
Gen(.)	Fuzzy generator function
Rep(.)	Fuzzy reproduction

master key K1, and computes L = h(K1||K2) and sa = K2 ⊕ L. GWN stores {SIDy, L} in its memory and sends sa to Sy. Sy after receiving sa computes L = sa⊕ K2 and stores {SIDy, L} in its memory. The procedure for registering a sensor node is depicted in Fig. 2.

User registration: User selects his IDx, PWx and Bx. A random number r is selected by Ux along with an integer I which is passed to the fuzzy verifier as a parameter such that 24 ≤ i ≤ 28. Ux computes Hx = h(IDx||r), HPWx = h(IDx||PWx||r), (σx, τx) = Gen(Bx) and HIDx = h(IDx||HPWx||σx)modi. Then Ux sends {Hx, HIDx} to GWN. After receiving the parameters {Hx, HIDx}, GWN computes KGWN−U = h(Hx||K1) and B1 = KGWN−U ⊕HIDx. After storing B1, h(.) in SC. Then it sends SC to Ux through a private channel.

After getting SC, Ux calculates Qx = h(HIDx||HPWx||σx||r) modi and B2 = r⊕h(IDx||PWx||σx) modi. Ux then stores {τx, Qx, B2} into SC so that SC = {B1, B2, τx, Qx, h(.)}. User registration process is outlined in Fig. 3.

Authentication and Key Agreement Phase

1. Ux places his smart card (SC) inthe reader and provides his IDx, PWx and Bx, and computes σx = Rep(Bx, τx), r = B2 ⊕ h(IDx||PWx||σx)mod i, HPWx = h(IDx||PWx||r), HIDx = h(IDx||HPWx||σx)mod i, Hx = h(IDx||r) and Q_x^* = h(HIDx||HPWx||σx||r)mod i. If $Q_x^* = Q_x$, then he computes KGWN−U = B1 ⊕HIDx, and generates timestamp TS1 and a random number gx. Then he computes B3 = Hx⊕ TS1, B4 = h(KGWN−U||Hx|| gx||TS1) and E1 = EnKGWN−U (B4||Hx||gx|| TS1). Ux then sends B3, E1 and TS1 to the GWN.
2. GWN first checks if $|TS_1 - TS_1'| < \Delta T$, where TS_1' is current time.If it holds true then GWN computes Hx = B3 ⊕ TS1, KGWN−U =h(Hx||K1), (B4||Hx||gx||TS1)= DecGWN−U (E1) and B_4^* =h(KGWN−U||Hx||gx||TS1). GWN verifies if $B_4^* = B4$. In case of equality, it generates timestamp TS2 and a random number

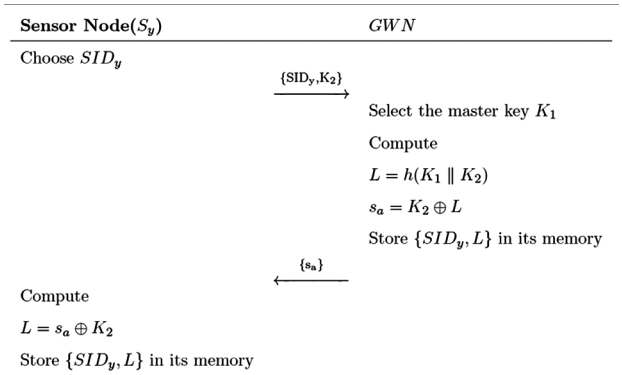


Fig. 2 — Sensor node registration phase

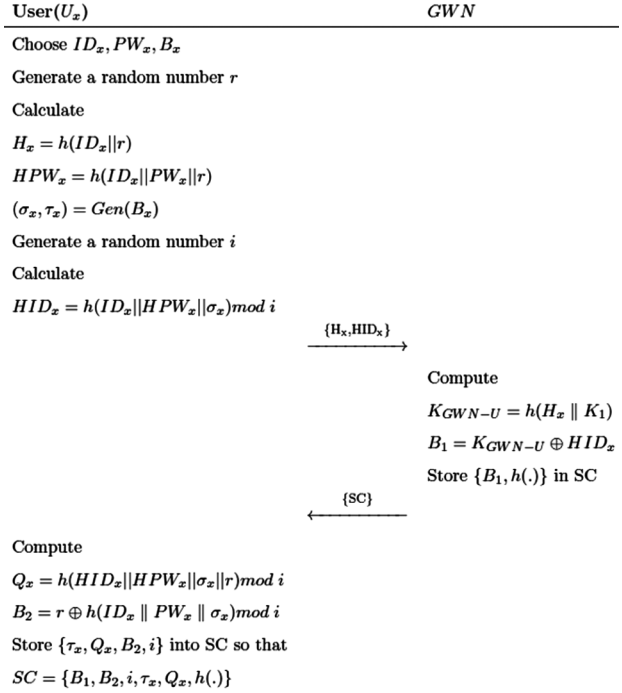


Fig. 3 — User registration phase

hx. It computes B5 = h(L||SIDy||TS1||TS2||hx) and B6 = EnL(B5||TS1||TS2|| Hx||gx||hx). GWN then sends B6 and TS2 to the Sy.

3. At time $TS_2^{'}$, Sy tests the condition $|TS_2 - TS_2^{'}| < \Delta T$; If it fails, the process is terminated. Else, it computes(B5||TS1||TS2||Hx||gx||hx)=DecL(B6)an dB$_5^*$ =h(L||SIDy||TS1||TS2||hx). It checks if B_5^* = B5. If this condition is hold, then it generates TS3 and a random number ix. It computes D1 = B5 ⊕ix, Skey_s = h(Hx||SIDy||ix||gx||hx) and B7 = h (Skey_s|| ix||gx||hx). Sy sends B7, D1 and TS3 to theGWN.
4. At time $TS_3^{'}$, GWN tests the condition $|TS_3 - TS_3^{'}| < \Delta T$; If it fails, the process is terminated. Else, it computes ix = B5 ⊕ D1 and Skey_g = h(Hx||SIDy||ix||gx||hx). If B7 = h(Skey-g||ix||gx||hx), then it generates timestamp TS4 and computesB8 =EnKGWN−U (Skey_g||SIDy||ix||gx||hx). GWN then sends B8 and TS4 to Ux.
5. At time $TS_4^{'}$, Uxtests the condition $|TS_4 - TS_4^{'}| < \Delta T$; If it is satisfied, then it derives ($S_{key_g}^{'}$||SIDy|| ix||gx||hx) = DecKGWN−U (B8) and Skey_ u = h(Hx||SIDy|| ix||gx||hx). If Skey_u = $S_{key_g}^{'}$, it generates TS5 and computes ACK = h(ix||Skey_u||gx|| TS5). Then it sends ACK and TS5 to Sy.
6. At time $TS_5^{'}$, Sy tests the condition $|TS_5 - TS_5^{'}| < \Delta T$; If it is satisfied, it checks ACK* = h(ix||Skey_s||gx|| TS5).

Authentication and Key Agreement phase is depicted in Fig. 4.

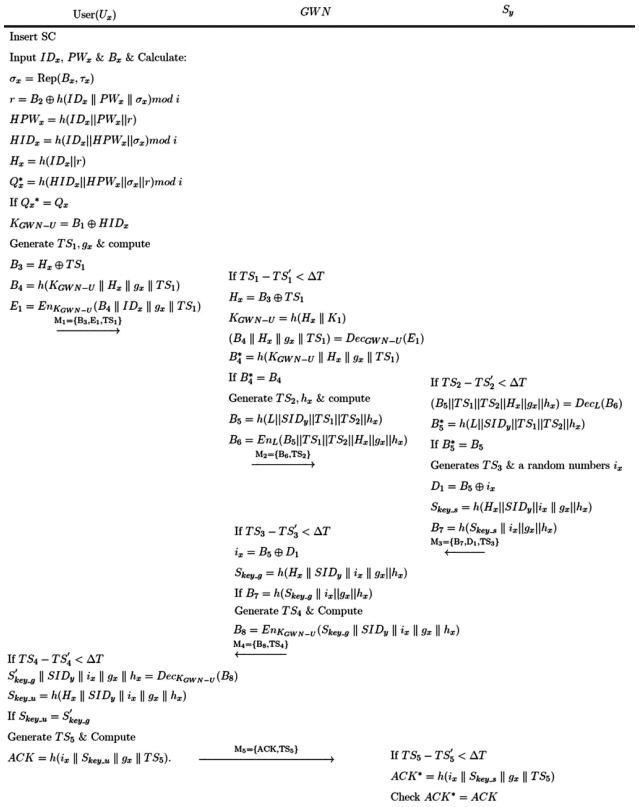


Fig. 4 — Authentication and key agreement phase

Password Update Phase

For updating their password, Ux executes the steps as given below:

1. To begin, Ux places the SC into the SC reader. The reader then prompts for the old identity IDx, password PWx and Bx.
2. Ux computes σx = Rep(Bx, τx), r = B2 ⊕ h(IDx||PWx||σx)mod i, HPWx = h(IDx||PWx||r), HIDx = h(IDx||HPWx||σx)mod i and Q_x^* = h(HIDx||HPWx||σx||r)mod i.
3. If $Q_x^* = Q_x$, then SC asks Ux to enter a new PW_x^{new} and evaluates KGWN−U = B1 ⊕HIDx.

User Ux enters the new password PW_x^{new} and derives HPWx = h(IDx||PW_x^{new}||r), (σx, τx) = Gen(Bx), HIDx =h(IDx||HPWx||σx)mod i, Q_x^{new}= h(HIDx||HPWx||σx||r)mod i, B_1^{new} = KGWN−U ⊕HIDx and B_2^{new}= r ⊕h(IDx|| PW_x^{new}|| σx)mod i.

Uxreplaces{Qx, B1, B2} with {Q_x^{new}, B_1^{new}, B_2^{new}} into SC, respectively.

Automated Security Verification via ProVerif

ProVerif is a popular open-source software commonly employed in the automatic formal verification process of security features of cryptographic protocols. As noted in schemes 31–34, it is primarily used for analyzing complex protocols and detect potential vulnerabilities.[31–34] This tool was employed to validate our proposed protocol's security. The results obtained after simulating the protocol in this tool are illustrated in Fig. 5 which confirms that the adversary cannot acquire the parameters {IDx, Skey_u, Skey_s, Skey_g} as the proposed authentication scheme proceeds through its defined phases in sequence.

Informal Assessment of Security Properties

This section assesses how well the proposed protocol withstands various security threats through a non-automated, reasoning-based

```
--Query not attacker(SKey_u[])
Completing...
Starting query not attacker(SKey_u[])
RESULT not attacker(SKey_u[]) is true.
--Query not attacker(SKey_g[])
Completing...
Starting query not attacker(SKey_g[])
RESULT not attacker(SKey_g[]) is true.
--Query not attacker(SKey_s[])
Completing...
Starting query not attacker(SKey_s[])
RESULT not attacker(SKey_s[]) is true.
--Query not attacker(IDx[])
Completing...
Starting query not attacker(IDx[])
RESULT not attacker(IDx[]) is true.
RESULT inj-event(UserAuthed)==>inj-event(UserStarted) is true.
RESULT inj-event(SensorAcGWN)==>inj-event(GWNAcceptUser)is true.
RESULT inj-event(GWNAcceptSensor)==>inj-event(SensorAcGWN) is true.
RESULT inj-event(UserAcceptGWN)==>inj-event(GWNAcceptSensor) is true.
```

Fig. 5 — Results obtained

evaluation commonly referred to as informal security analysis.

Replay Attack

Here, the intruder captures the message M1 = {B3, E1, TS1} and attempts to replay it to perform other attacks. But GWN detects this attack becauseGWN first checks whether $|TS_1 - TS_1'| < \Delta T$ or not. This inequality does nothold because timestamphas changed. Then it checks if B4 = h(KGWN−U||Hx||gx|| TS1), if not true then GWN will abort it.

Similarly, if M2 = {B6, TS2} is replayed, then also the Sy will detect this because it first checks whether $|TS_2 - TS_2'| < \Delta T$ as timestamphas changed so this will not hold true. Then it checks whetherB5=h(L||SIDy||TS1||TS2||hx). If not true then Sy will abort it.

If message M3 = {B7, D1, TS3} is replayed then GWN will detect this, because initially it checks whether $|TS_3 - TS_3'| < \Delta T$ because the timestamphas changed so this will not hold true. Then it checks B7 = h(Skey_g||ix||gx||hx), if not true then GWN will abort it.

If message M4 = {B8, TS4} is replayed then Ux will find the attackbecause it initially checks whether $|TS_4 - TS_4'| < \Delta T$, this will not hold truebecause of the change in timestamp. Then it checks whether S key_ u = S'_{key_g}, if not true then Ux will abort it.

Similarly, if message M5 = {ACK, TS5} is replayed then Sy will detectthis because Sy initially checks whether $|TS_5 - TS_5'| < \Delta T$, this will nothold because of the change in timestamp. Then it checks whether ACK = h(ix||Skey_s||gx||TS5), if not true then Sy will abort it. Therefore, our improved protocol resists the replay attack.

Mutual Authentication

Mutual authentication is a processs which required the two participating ends authenticate each other before starting actual communication. Our improved protocol guarantees mutual authentication. In login and authentication phase, user Ux enters IDx, PWx, Bx and checks whether Qx = h(HIDx||HPWx||σx||r)mod i, if true then only login request is made, otherwise abort it. GWN only authenticates Ux if B4 = h(KGWN−U||Hx||gx|| TS1), otherwise abort it. Sy checks the authenticity by verifying whether B5= h(L||SIDy||TS1||TS2||hx), only Sy has the secret parameter hx, ix and TS3 and can compute Skey_s = h(Hx||SIDy||ix||gx||hx). Thus, Ux verifies GWN and Sy by checking whether Skey_u = Skey_g

Untraceability

It is a feature which ensures that an attacker cannot identify Ux in different sessions. Ux sends the message M1 = {B3, E1, TS1} to GWN, where B3 = Hx⊕ TS1, Hx = h(IDx||r) and E1 = EnKGWN−U (B4||IDx||gx||TS1). B3 is safe as Hx is protected using hash function and E1 is encrypted using symmetric key cryptography. In each session, value of gx and TS1 are different, therefore attacker is not able to trace Ux.

Privileged insider attack

Even if the adversary correctly guesses the PWx and IDx, and further obtains the biometrics Bx, still he cannot verify the message Qx = h(HIDx||HPWx||σx||r)mod i because it is secured using fuzzy verifier i.

Perfect forward security

It makes sure that session key will remain safe even if the secret components used in the session key exchange gets exposed. Here, Skey_s = h(Hx||SIDy||ix||gx||hx), which is protected random numbers ix, gx and hx and by hash function.

Session key security

As explained in the previous security analysis also, Skey_s = h(Hx||SIDy||ix||gx||hx), which is protected

Table 2 — Security features

Security feature	Das[28]	Qi *et al.*[29]	Kumari *et al.*[30]	Know *et al.*[35]	Proposed
Three-factor security	—	—	—	—	✓
Session key security	×	×	×	×	✓
Perfect forward secrecy	✓	✓	✓	×	✓
Mutual authentication	✓	✓	✓	✓	✓
Password guessing attack	✓	✓	✓	×	✓
Privileged insider attack	✓	✓	×	×	✓
Replay attack	✓	✓	✓	✓	✓
Untraceability	×	×	×	×	✓
User impersonation attack	✓	✓	✓	✓	✓
Device stolen attack	✓	×	✓	×	✓

by hash function and random numbers ix, gx and hx. Hence, session key remains secured from an attack.

Device stolen attack

In case the attacker stole the device, gets the SC and may get the PWx, IDx, and extract Qx = h(HIDx||HPWx||σx||r)mod i but will fail to verify message validity because PWx and IDx are protected using fuzzy verifier i.

User Impersonation Attack

An intruder might attempt to fabricate the login data however such an attempt remains unsuccessful even if it intercepts the message M1 = {B3, E1, TS1}, where B3 = Hx⊕TS1, Hx = h(IDx||r) and E1 = EnKGWN−U (B4||IDx||gx||TS1), which are protected through the combined use of hashing and symmetric key encryption.

Password Guessing Attack

Even if the adversary correctly guesses PWx and obtain other information yet he/she still would not be able to verify if this guess is correct because Qx = h(HIDx||HPWx||σx||r)mod i, where PWx is protected by a fuzzy verifier.

MITM Attack

At each stage, the mechanism firstly authenticates Ux to GWN, followed by authentication of GWN to Sy thereby achieving mutual authentication. Therefore, it resists MITM attack.

Three-Factor Security

Smart Card, biometrics and password (three factors) guarantee that session key cannot be revealed even if the two of the secret components used gets exposed.

Performance Analysis

In this section, the proposed protocol is verified against related protocols[28–30,35], focusing on three aspects: security features, computational cost, and communication cost.

Security Features

A comparison of the security features between the proposed scheme and existing protocols[28–30,35] is presented in Table 2. This table shows that the protocols 28–30, 35 fail to provide untraceability and session key security. The protocols 30,35 cannot resist privileged insider attack.[30,35] The schemes 29,35 fail to provide resistance against device stolen attack.[29,35] Although the protocols 28–30,35 can resist replay attack and provide mutual authentication but they are vulnerable to other attacks. However, the proposed protocol is able to provide all the security measures. This highlights the superior robustness and security of the scheme proposed in this work w. r. t. the existing related other schemes.

Here, ✓ represents that the protocol resists the attack, × represents that it fails to do so and —represents that the feature is not defined for the protocol.

Computational Cost

The total time taken to execute a protocol is referred to as its computation cost. To assess efficiency, this study evaluated and compared the computational cost of the proposed scheme against related protocols[28–30,35] in Table 3. T_h represents the time taken for hash operations, T_f is the run time of the fuzzy extractor, T_{cm} corresponds to the execution time of a chaotic map function, and T_s denotes the time required for encryption/decryption operations. As concatenation and XOR operation have negligible computational cost, it can be ignored. For computation cost calculation, only the authentication phase is considered because it is executed most frequently.

For comparison of the execution time of the current study with protocols of the related literature[28–30,35] the same cryptographic operation timings as specified by

Table 3 — Cost comparison

Entities	Das[28]	Qi *et al.*[29]	Kumari *et al.*[30]	Know *et al.*[35]	Proposed
User	$6T_h+T_s+2T_f$	$7T_h+4T_{cm}+T_f$	$5T_h+2T_s+3T_{cm}$	$12T_h$	$5T_h+T_s+T_f$
Gateway	$3T_h+2T_s$	$6T_h+T_s+T_{cm}$	$6T_h+T_s$	$22T_h$	$6T_h+4T_s$
Sensor	$2T_h+T_s$	$4T_h+T_s+5T_{cm}$	$3T_h+3T_{cm}$	$10T_h$	$6T_h+T_s$
Total	$11T_h+4T_s+2T_f$	$17T_h+2T_s+10T_{cm}+T_f$	$14T_h+3T_s+6T_{cm}$	$44T_h$	$17T_h+6T_s+T_f$
Exec.time	$0.0747s$	$0.2141s$	$0.13585s$	0.022s	$0.0781s$
Data used (Bits)	1152	1880	1568	3392	2144

Chen & Chen[36] was considered. So T_h = 0.0005s, T_s = 0.00875s, T_f = 0.0171s, and T_{cm} = 0.0171s. From Table 3 it is clear that execution time of the proposed protocol is less than the protocols of Qi *et al.*[29] and Kumari *et al.*[30]) but higher than the schemes of Das[28] and Know *et al.*[35] However, the additional computational cost is justified due to supporting more security features by the proposed protocol. Unlike the compared schemes, the proposed scheme achieves all the security properties listed in Table 2. While the protocol by Das[28] lacks untraceability and session key security, the weaknesses of the protocol of Know *et al.*[35] are more extensive. It does not ensure untraceability, leaves the session key unprotected, and fails to guarantee perfect forward secrecy. In addition, it remains vulnerable to privileged insider attack, device theft, and offline password guessing. Since authentication in IIoT is performed only when a user accesses the system, the slight increase in computation is acceptable in exchange for stronger security guarantees.

Communication Cost

Total amount of data, in bits, transmitted between the participants is used to compute the communication cost. Comparison of the communication cost of the proposed scheme with that of the associated schemes[28–30,35] is given in Table 3. For this computation, both the encryption/decryption operations and the hash operation are assumed to be 256 bits each, the timestamp is taken as 32 bits, and the identity and the random number/nonce are assumed to be 160 bits each. The results in Table 3 show that the proposed scheme incurs a lower communication cost than Know *et al.*[35], though a higher cost compared to the protocols of the remaining studies.[28–30] This increase in communication cost is modest and is justified by the additional security properties achieved by the proposed protocol, including untraceability, session key security, resistance to device stolen attacks, and support for three-factor authentication, as summarized in Table 2. In real-world IIoT deployments, authentication messages are exchanged only during session establishment, whereas secure communication continues for a significantly longer duration. Therefore, the slight increase in communication cost has a negligible effect on overall system performance while ensuring substantially stronger security guarantees.

Conclusions

Security remains a critical requirement in wireless sensor networks (WSNs) deployed in Industry 4.0 environments. The current study has designed and presented a refined authentication scheme conceptualized using symmetric key cryptography tailored for resource constrained IIoT domain. Simple operations like XOR and hashing have been used to minimize the computation cost. Its security is verified formally using ProVerif tool. Also, informal security analysis demonstrated that the scheme proposed here effectively resists a wide range of attacks. Furthermore, performance analysis revealed that the protocol presented here provides better security in comparison to the other associated protocols. Future studies will focus on reducing the computation and communication cost.

References

1 Das M L, Two-factor user authentication in wireless sensor networks, *IEEE Trans Wireless Commun*, **8(3)** (2009) 1086–1090, doi: 10.1109/TWC.2008.080128.

2 He D, Gao Y, Chan S, Chen C & Bu J, An enhanced two-factor user authentication scheme in wireless sensor networks, *Ad Hoc Sens Wirel Netw*, **10(4)** (2010) 361–371.

3 Khan M K & Alghathbar K, Cryptanalysis and security improvements of a two-factor user authentication in wireless sensor networks, *Sensors*, **10(3)** (2010) 2450–2459, doi: 10.3390/s100302450.

4 Yoon E J & Yoo K Y, Cryptanalysis of robust mutual authentication protocol for wireless sensor networks, *Proc 10th IEEE Int Conf Cogn Inform Cogn Comput (ICCI*CC 2011)*, (2011) 392–396, doi: 10.1109/COGINF.2011.6016171.

5 He D, Robust biometric-based user authentication scheme for wireless sensor networks, *IACR Cryptologye Print Archive*, (2012) 203.

6 Kim J, Lee D, Jeon W, Lee Y & Won D, Security analysis and improvements of two-factor mutual authentication with key agreement in wireless sensor networks, *Sensors*, **14(4)** (2014) 6443–6462, doi: 10.3390/s140406443.

7 Chang I P, Lee T F, Lin T H & Liu C M, Enhanced two-factor authentication and key agreement using dynamic

identities in wireless sensor networks, *Sensors*, **15(12)** (2015) 29841–29854, doi: 10.3390/s151229767.

8 Farash M S, Turkanović M, Kumari S & Hölbl M, An efficient user authentication and key agreement scheme for heterogeneous wireless sensor network tailored for the internet of things environment, *Ad Hoc Netw*, **36** (2016) 152–176, doi: 10.1016/j.adhoc.2015.05.014.

9 Amin R, Islam S H, Biswas G, Khan M K, Leng L & Kumar N, Design of an anonymity-preserving three-factor authenticated key exchange protocol for wireless sensor networks, *Comput Netw*, **101** (2016) 42–62, doi: 10.1016/j.comnet.2016.01.006.

10 Park Y & Park Y, Three-factor user authentication and key agreement using elliptic curve cryptosystem in wireless sensor networks, *Sensors*, **16(12)** (2016) 2123, doi: 10.3390/s16122123.

11 Amin R & Biswas G, A secure light weight scheme for user authentication and key agreement in multi-gateway based wireless sensor networks, *Ad Hoc Netw*, **36** (2016) 58–80, doi: 10.1016/j.adhoc.2015.05.020.

12 Lu Y, Li L, Peng H & Yang Y, An energy efficient mutual authentication and key agreement scheme preserving anonymity for wireless sensor networks, *Sensors*, **16(6)** (2016) 837, doi: 10.3390/s16060837.

13 Lee C C, Li C T, Chiu S T & Lai Y M, A new three-party-authenticated key agreement scheme based on chaotic maps without password table, *Nonlinear Dyn*, **79(4)** (2015) 2485–2495, doi: 10.1007/s11071-014-1827-x.

14 Jabbari A & Mohasefi J, Improvement in new three-party-authenticated key agreement scheme based on chaotic maps without password table, *Nonlinear Dyn*, **95(4)** (2019) 3177–3191, doi: 10.1007/s11071-018-04748-y.

15 Tewari A & Gupta B B, Cryptanalysis of a novel ultra-lightweight mutual authentication protocol for IoT devices using RFID tags, *J Supercomput*, **73(3)** (2017) 1085–1102, doi: 10.1007/s11227-016-1849-x.

16 Li J, Zhang W, Kumari S, Choo K K R & Hogrefe D, Security analysis and improvement of a mutual authentication and key agreement solution for wireless sensor networks using chaotic maps, *Trans Emerg Telecommun Technol*, **29(15)** (2018) e3295, doi: 10.1002/ett.3295.

17 Das A K, Wazid M, Kumar N, Vasilakos A V & Rodrigues J J P C, Biometrics-based privacy-preserving user authentication scheme for cloud-based industrial internet of things deployment, *IEEE Internet Things J*, **5(6)** (2018) 4900–4913, doi: 10.1109/JIOT.2018.2877690.

18 Hussain S & Chaudhry S A, Comments on a biometrics-based privacy-preserving user authentication scheme for cloud-based industrial internet of things deployment, *IEEE Internet Things J*, **6(6)** (2019) 10936–10940, doi: 10.1109/JIOT.2019.2934947.

19 Kaur D, Kumar D, Saini K K & Grover H S, An improved user authentication protocol for wireless sensor networks, *Trans Emerg Telecommun Technol*, **30(10)** (2019) e3745, doi: 10.1002/ett.3745.

20 Kumar D, Grover H S, Kaur D, Verma A, Saini K K & Kumar B, An efficient anonymous user authentication and key agreement protocol for wireless sensor networks, *Int J Commun Syst*, **34(5)** (2021) e4724, doi: 10.1002/dac.4724.

21 Gope P & Hwang T, A realistic lightweight anonymous authentication protocol for securing real-time application data access in wireless sensor networks, *IEEE Trans Ind Electron*, **63(11)** (2016) 7124–7132, doi: 10.1109/TIE.2016.2585081.

22 Kumar D, A secure and efficient user authentication protocol for wireless sensor network, *Multimed Tools Appl*, **80(1)** (2021) 27131–27154, doi: 10.1007/s11042-021-10950-9.

23 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** (2022) 12137–12152, doi: 10.1109/ACCESS.2022.3146393.

24 Neerul & Kumar D, Cryptanalysis and improvement of an authentication protocol for wireless sensor networks, *Trans Emerg Telecommun Technol*, **34(5)** (2023) e4747, doi: 10.1002/ett.4747.

25 Dolev D & Yao A C, On the security of public key protocols, *IEEE Trans Inf Theory*, **29(2)** (1983) 198–208, doi: 10.1109/TIT.1983.1056650.

26 Kumar D, Chand S & Kumar B, Cryptanalysis and improvement of a two-factor user authentication scheme, *Infocommunications J*, **10(1)** (2018) 10–14, doi: 10.36244/ICJ.2018.1.2.

27 Wang D & Wang P, Two birds with one stone: Two-factor authentication with security beyond conventional bound, *IEEE Trans Dependable Secur Comput*, **15** (2018) 708–722, doi: 10.1109/TDSC.2016.2605087.

28 Das A K, A secure and effective biometric-based user authentication scheme for wireless sensor networks using smart card and fuzzy extractor, *Int J Commun Syst*, **30(1)** (2017) e2933, doi: 10.1002/dac.2933.

29 Qi M, Chen J & Chen Y, A secure biometrics-based authentication key exchange protocol for multi-server TMIS using ECC, *Comput Methods Programs Biomed*, **164** (2018) 101–109, doi: 10.1016/j.cmpb.2018.07.008.

30 Kumari S, Li X, Wu F, Das A K, Arshad H & Khan M K, A user friendly mutual authentication and key agreement scheme for wireless sensor networks using chaotic maps, *Future Gener Comput Syst*, **63** (2016) 56–75, doi: 10.1016/j.future.2016.04.016.

31 Xiong H, Wu Y, Jin C & Kumari S, Efficient and privacy-preserving authentication protocol for heterogeneous systems in IIoT, *IEEE Internet Things J*, **7(12)** (2020) 11713–11724, doi: 10.1109/JIOT.2020.2999510.

32 Wu T Y, Lee Z, Obaidat M S, Kumari S, Kumar S & Chen C M, An authenticated key exchange protocol for multi-server architecture in 5G networks, *IEEE Access*, **8** (2020) 28096–28108, doi: 10.1109/ACCESS.2020.2969986.

33 Blanchet B, Smyth B, Cheval V & Sylvestre M, ProVerif 2.00: Automatic cryptographic protocol verifier, user manual and tutorial, (2018).

34 Wu T Y, Yang L, Lee Z, Chen C M, Pan J S & Islam S H, Improved ECC-based three-factor multiserver authentication scheme, *Secur Commun Networks*, **2021(1)** (2021) 6627956, doi: 10.1155/2021/6627956.

35 Kwon D K, Yu S J, Lee J Y, Son S H & Park Y H, WSN-SLAP: Secure and lightweight mutual authentication protocol for wireless sensor networks, *Sensors*, **21(3)** (2021) 936, doi: 10.3390/s21030936.

36 Chen Y & Chen J, Anonymous and provably secure authentication protocol using self-certified cryptography for wireless sensor networks, *Multimed Tools Appl*, **80** (2021) 15291–15313, doi: 10.1007/s11042-020-10259-z.