A Lightweight Anonymous Authentication and Key Agreement Protocol for Wearable Devices Using PUF

Fangchen Xu[1], Juncheng Tong[2], Huijie Yang[1], Xinyu Li[3], and Quanrun Li[1]

[1] School of Information Science and Engineering(School of Cyber Science and Technology), Zhejiang Sci-Tech University, Hangzhou, China
[2] School of cyber science and engineering,Wuhan University
jctong@whu.edu.cn
[3] Hubei Key Laboratory of Intelligent Robot,Wuhan Institute of Technology,430250,Wuhan,China
xinyuliwhu@whu.edu.cn

Abstract. With the widespread deployment of wearable devices in scenarios such as smart healthcare and health monitoring, the security issues of identity authentication and key agreement faced by them in open wireless environments have become increasingly prominent. Constrained by computational capabilities and storage resources, traditional security protocols relying on complex cryptographic operations are unsuitable for wearable devices, while many lightweight schemes suffer from insufficient security. Especially in the three-party collaborative architecture of wearable device - mobile terminal - cloud server, the cloud server can often obtain or deduce the session keys between terminals. To address the above problems, this paper presents a lightweight anonymous authentication and key agreement protocol using Physical Unclonable Function. This scheme generates secure session keys for both users and wearable devices, and users and cloud servers within a single authentication, and ensures that the final session key between wearable devices and mobile terminals remains invisible to cloud servers, thereby achieving end-to-end key privacy protection. This paper conducts a formal security proof of the proposed scheme under the ROR model. Performance analysis shows that compared with related schemes, the protocol achieves better performance with regard to computational and communication overheads, and is suitable for resource-limited wearable device application scenarios.

Keywords: Wearable devices · Lightweight key agreement · Physical Unclonable Function · Anonymous authentication · ROR model.

1 Introduction

As mobile computing and sensing technologies improve, wearable devices have gradually evolved from independent electronic products into intelligent terminals that can be embedded in clothing, accessories, and even skin patches. Devices

such as smartwatches, fitness trackers are now widely used in scenarios like health monitoring, immersive experiences[24]. Relying on high-precision sensors and low-power communication modules, these devices can continuously record users' physiological and behavioral data, which are typically forwarded to cloud servers via mobile terminals for further analysis and processing[10].

However, the popularization of wearable systems is accompanied by increasingly prominent security risks. Limited by hardware resources, wearable devices cannot support complex security algorithms[3]; and their reliance on wireless communication links makes them vulnerable to eavesdropping, spoofing, and replay attacks. In addition, security vulnerabilities in the application programs, firmware updates, and data storage processes of wearable devices may all become attack entry points, exposing users' private data such as physiological information and activity trajectories to the risk of leakage[21].

Therefore, it is essential to design secure, reliable, and low-overhead communication and authentication mechanisms for wearable devices. In the existing three-party collaborative architecture, wearable devices are responsible for data collection, mobile terminals handle data processing, and trusted cloud servers are in charge of data storage and analysis, as well as assisting in completing identity-related verification and coordination tasks required for system operation[18]. However, due to the lack of sufficient computational capabilities of wearable devices themselves, they often rely on the collaborative participation of mobile terminals and trusted cloud servers during identity authentication and key agreement[19]. Consequently, designing a lightweight, highly secure key agreement protocol with privacy protection under the three-party architecture remains a challenging research problem.

In recent years, various authentication and key agreement schemes for wearable environments have been introduced, but many rely on complex cryptographic components, making them difficult to deploy on resource-constrained wearable devices, while some lightweight schemes suffer from insufficient security[25]. Against this backdrop, Physical Unclonable Function (PUF), which possess hardware-unique characteristics, have gradually become a reliable security primitive. PUF can generate stable and non-replicable responses based on minor differences in the chip manufacturing process, enabling them to serve both as lightweight identity credentials and provide strong security for protocols. Based on these features, PUF are particularly suitable for identity authentication and key establishment in wearable scenarios[22].

In existing wearable systems, the session key between a mobile terminal and a wearable device is often generated by the cloud server or is visible in the cloud, which may lead to key privacy leakage. To address this issue, the proposed protocol ensures that the cloud server remains unaware of the session key between the device and terminal, thereby achieving end-to-end key privacy protection. This feature fills a gap in existing schemes, enhancing session key security[1]. The system architecture is presented in Fig. 1.

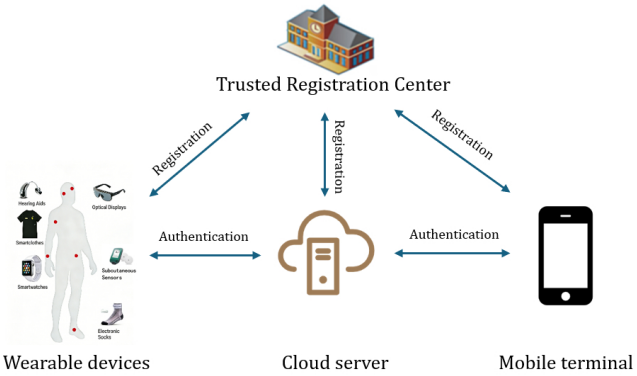

Fig. 1. system architecture

This paper presents a lightweight and secure key agreement protocol for wearable devices based on the features of PUFs, with the main contributions outlined below.

1. Presenting a lightweight and efficient protocol for anonymous authentication and key agreement, by using the uniqueness and unclonability of PUF to provide wearable devices with a unique identity authentication function. This protocol not only achieves mutual authentication between the three parties of wearable devices, mobile terminals and trusted cloud servers, but also generates two independent session keys (user-wearable device and user-cloud server) simultaneously in one authentication process, thus significantly improving the key establishment efficiency and overall security in multi-entity collaboration scenarios.
2. The proposed protocol ensures that the cloud server cannot acquire the final session key between the wearable device and mobile terminal, thereby enhancing end-to-end security and avoiding the risk of single-point failure at the cloud server.
3. The protocol's security is validated using the ROR model, with formal analysis confirming its provable security.
4. Compared to existing protocols, this scheme offers improved security and better efficiency, with lower computational and communication overheads, especially in large-scale wearable networks.

2 RELATED WORK

In recent years, various authentication and key agreement protocols for wearable environments have emerged; however, most studies have failed to effectively balance the performance requirements of resource-constrained devices with the reliability of secure communication. Specifically, Qi et al[11].proposed an authentication protocol using quadratic residue theory, aiming to ensure secure

communication between wearable devices and healthcare personnel; Challa et al[4].proposed a three-factor authentication protocol supporting password updates. Although these schemes have been proven secure, they generally suffer from high computational complexity, making it difficult to adapt to the limited processing capabilities and resource constraints of wearable devices.

To address the challenge of resource constraints, researchers have shifted their focus to the design of lightweight schemes. Mahdi et al[8].designed a novel lightweight hash chain-based scheme; however, Chen et al[5].indicated this protocol is susceptible to device capture attacks and lacks mutual authentication. The authentication scheme proposed by Wu et al[23]. achieves key agreement solely relying on hash functions and XOR operations while ensuring device anonymity; yet Lee et al[12].found that it suffers from insider attacks, device theft, and physical cloning attacks, and fails to ensure proper mutual authentication. Similarly, although Guo et al[9].proposed a lightweight anonymous authentication scheme aiming to support mutual authentication among multiple entities, research by Xin et al[1].demonstrates that it does not effectively achieve secure session key establishment.

Despite extensive research on authentication and key agreement for wearable devices, many existing schemes still struggle to satisfy practical requirements due to limited computational resources and stringent security and efficiency demands. To address this, PUF have been increasingly introduced as a hardware-based security primitive. PUFs rely on manufacturing variations to generate unique and unclonable identifiers, offering low computational overhead, low energy consumption, and strong resistance to cloning and physical tampering, making them well suited for authentication and key generation in wearable devices. Since Suh et al[17].first applied PUFs to authentication and key generation, numerous PUF-based schemes have been introduced. Liu et al[14].proposed an ultra-lightweight authentication protocol based on PUFs, supported by the cloud, which realizes authentication between a smartphone and two wearable devices. Alruwaili et al[2].demonstrated the feasibility of PUF-based security mechanisms on resource-constrained wearable devices.

Although several PUF-based authentication and key agreement protocols have been introduced, in most works, the trusted center not only distributes key materials to wearable devices and mobile terminals but also can even derive or indirectly obtain the session key between the two parties under certain conditions. This high reliance on a centralized node exposes the system to potential single-point failure risks. The proposed protocol, unlike existing designs, maintains lightweight authentication and key agreement while ensuring that the session key is only generated and shared between wearable devices and mobile terminals, with the trusted cloud server remaining unaware of the key throughout the entire process. In addition, this protocol supports mutual authentication between wearable devices and users as well as between cloud servers and users, and is suitable for the three-party collaborative architecture of wearable device-mobile terminal-trusted center.

3 PRELIMINARIES

3.1 Physical Unclonable Functions

Physical Unclonable Function (PUF) is a function based on the microscopic hardware variations of a device, generating a unique identity through a challenge-response mechanism. PUF ensures that each device produces a consistent response to the same challenge, guaranteeing identity authentication consistency. PUF, integrated within the device's physical structure, is hard to replicate or clone, providing strong physical unclonability and unidirectionality, meaning that even if a response is obtained, the corresponding challenge cannot be easily deduced. PUFs are widely used for device authentication and key generation, serving as a digital fingerprint to prevent physical attacks.[15].

3.2 Fuzzy Extractor

A Fuzzy Extractor is a cryptographic construct designed for noisy inputs, enabling the stable generation of a consistent random string from imperfectly repeatable data. Its core idea is to map inputs that are "sufficiently close" to the same key while maintaining security, without requiring exact input replication. The Fuzzy Extractor includes two algorithms: a generation algorithm, which outputs a high-entropy random string and a piece of public auxiliary data when it first acquires the input, and a recovery algorithm, which reconstructs the same string from a slightly deviated new input using the auxiliary data. The auxiliary data reveals no information about the key and is used solely to correct input noise[6].Its tolerance for input errors makes it particularly suitable for key generation in noisy environments, such as biometric recognition systems.

4 SYSTEM MODEL AND THREAT MODEL

4.1 System Model

In the wearable computing system model designed in this paper, the system consists of four entities: wearable devices, users/mobile terminals, a Trusted Registration Center (RC), and a trusted cloud server (CS). Each user is equipped with a mobile terminal and multiple wearable devices for data collection. Wearable devices transmit data wirelessly to the terminal, and the mobile terminal processes and uploads it to the trusted cloud server. The RC securely distributes credentials to devices and terminals in an offline manner, enabling effective identity management, and stores only the credentials necessary for authentication in the CS to support device and terminal verification. The CS is responsible for data storage and processing, ensuring data integrity and privacy, while authorized users can access and analyze the data. This model secures data during transmission and storage, optimizes authentication and data management, and supports timely and accurate monitoring of user information.

4.2 Adversary Model

This study applies the Dolev-Yao (DY) model[7] to evaluate the protocol's security, assuming adversary $\mathcal{A}$ controls the communication channel and is capable of launching attack operations such as eavesdropping, tampering, forgery, deletion, and replay of messages. The communication process takes place over an insecure channel, where the adversary can intercept and modify messages but cannot decrypt encrypted content. The adversary can only decrypt or sign messages when in possession of the correct key. Meanwhile, the adversary may physically capture legitimate devices in communication and extract information stored in the devices, but it cannot perform cryptanalytic operations to crack the keys.

Assuming the CS is equipped with strong anti-tampering measures and sufficient computing power to ensure the secure storage and processing of sensitive information, the terminal devices, due to their limited storage and computational capabilities, are more vulnerable to attacks, especially session key query and participant compromise attacks.

5 PROPOSED SCHEME

This section describes in detail the four phases of the proposed protocol: (1) Wearable Device Registration, (2) Mobile Terminal Registration, (3) Login, and(4) Authentication and Key Agreement. The symbols and their associated definitions are listed in Table 1.

Table 1. Notation and Description

Notation	Description
RC	Trusted registration center
CS	Cloud server
MT_i	Mobile terminal
ID_{WD_j}	Identity of WD_j
PID_{WD_j}	Pseudo-identity of WD_j
TID_{WD_j}	Temporary identity of WD_j
$\langle C_i, R_i \rangle$	The Challenge-Response pair of PUF
ID_i	Identity of the i^{th} user
PID_i	Pseudo-identity of ID_i
TID_i	Temporary identity of ID_i
BIO_i	Biometrics of the i^{th} user
$\mathrm{Gen}(\cdot)$	Fuzzy extractor obfuscation function
$\mathrm{Rep}(\cdot)$	Fuzzy extractor reproduction function
$h(\cdot)$	Hash function
$\oplus$, $\|$	Bitwise XOR operation, Concatenation operation

5.1 Registration of Wearable Devices

(1) RC sends the identity ID_{WD_j} to WD_j.
(2) After WD_j receives it, it uses C_i to stimulate the PUF, generates $R_i = \text{PUF}(C_i)$, and sends ID_{WD_j} and R_i to RC.
(3) After RC receives them, it selects a random number r, calculates WD_j's pseudonym $PID_{WD_j} = h(ID_{WD_j} \parallel r)$ and temporary pseudonym $TID_{WD_j} = h(ID_{WD_j} \parallel R_i \parallel T_1)$, sends PID_{WD_j} to WD_j, and records $< ID_{WD_j}, R_i, TID_{WD_j} >$ in CS's database; WD_j saves $\{ID_{WD_j}, TID_{WD_j}, PID_{WD_j}, C_i\}$ locally.

5.2 Registration of Mobile Terminals

(1) Via MT_i, the user enters the unique identity ID_i, the corresponding password PW_i, and the biometric feature BIO_i.
MT_i then selects a random number n_1, calculates the pseudonym $PID_i = h(ID_i \parallel n_1)$ and the pseudo-password $PPW_i = h(PW_i \parallel n_1)$, as well as $(k_i, hid_i) = \text{Gen}(BIO_i)$, and sends $\{ID_i, PID_i, PPW_i, k_i\}$ to RC.
(2) After RC receives them, it selects a random number r, calculates $a_i = r \oplus h(PID_i \parallel PPW_i \parallel k_i)$ and the temporary pseudonym $TID_i = h(ID_i \parallel k_i \parallel T_2)$, sends $\{a_i, TID_i\}$ to MT_i, and records the information $< ID_i, k_i, TID_i >$ in CS's database.
(3) MT_i calculates the token $Auth_i = h(PID_i \parallel PPW_i \parallel k_i \parallel n_1 \parallel r)$ and $b_i = n_1 \oplus h(ID_i \parallel PW_i \parallel k_i)$, then saves $\{a_i, TID_i, Auth_i, b_i\}$.

5.3 User Login

When logging in, the user enters their own ID_i', PW_i', and biometric feature BIO_i', recovers $k_i' = \text{Rep}(hid_i, BIO_i')$. MT_i calculates $n_1' = b_i \oplus h(ID_i' \parallel PW_i' \parallel k_i')$, $PID_i' = h(ID_i' \parallel n_1')$, $PPW_i' = h(PW_i' \parallel n_1')$, and $r' = a_i \oplus h(PID_i' \parallel PPW_i' \parallel k_i')$. Finally, it verifies the token $Auth_i' = h(PID_i' \parallel PPW_i' \parallel k_i' \parallel n_1' \parallel r')$; if the verification succeeds, the user logs in successfully.

5.4 Authentication and Key Agreement

(1) MT_i selects a random number λ and a timestamp t_1, calculates $w_i = \lambda \oplus h(ID_i \parallel k_i \parallel t_1)$ and $M_1 = h(TID_i \parallel \lambda \parallel ID_i \parallel k_i \parallel t_1)$. MT_i sends $\{TID_i, w_i, M_1, t_1\}$ to the trusted cloud server CS.
(2) CS first checks the freshness of timestamp t_1, retrieves the random numbers ID_i and k_i corresponding to TID_i, calculates $\lambda = w_i \oplus h(ID_i \parallel k_i \parallel t_1)$, and verifies $M_1 = h(TID_i \parallel \lambda \parallel ID_i \parallel k_i \parallel t_1)$. If the verification succeeds, it generates a random number α and a timestamp t_2, calculates $x = h(R_i \parallel ID_{WD_j} \parallel t_2) \oplus \alpha$ and $M_2 = h(TID_{WD_j} \parallel \alpha \parallel ID_{WD_j} \parallel R_i \parallel t_2)$, then sends $\{TID_{WD_j}, x, M_2, t_2\}$ to WD_j.
(3) WD_j checks the freshness of timestamp t_2, calculates $R_i = \text{PUF}(C_i)$, computes $\alpha = h(R_i \parallel ID_{WD_j} \parallel t_2) \oplus x$, and verifies $M_2 = h(TID_{WD_j} \parallel \alpha \parallel ID_{WD_j} \parallel R_i \parallel t_2)$. If the verification succeeds, it selects a random number

β and a timestamp t_3, calculates $m_j = h(\alpha \parallel R_i \parallel ID_{WD_j} \parallel t_3) \oplus \beta$ and $M_3 = h(\beta \parallel \alpha \parallel R_i \parallel ID_{WD_j} \parallel t_3)$, then sends $\{TID_{WD_j}, m_j, M_3, t_3\}$ to CS.
(4) CS checks the freshness of timestamp t_3, calculates $\beta = h(\alpha \parallel R_i \parallel ID_{WD_j} \parallel t_3) \oplus m_j$, and verifies $M_3 = h(\beta \parallel \alpha \parallel R_i \parallel ID_{WD_j} \parallel t_3)$. If the verification succeeds, it selects a random number φ and a timestamp t_4, calculates $y = \varphi \oplus h(ID_i \parallel k_i \parallel t_4)$, $z_1 = ID_{WD_j} \oplus \varphi$, $z_2 = \beta \oplus \varphi$, $sk_{CS-MT_i} = h(TID_i \parallel ID_i \parallel k_i \parallel t_4 \parallel \varphi)$, and $M_4 = h(TID_i \parallel ID_i \parallel TID_{WD_j} \parallel ID_{WD_j} \parallel \beta \parallel sk_{CS-MT_i} \parallel k_i \parallel t_4 \parallel \varphi)$, then sends $\{TID_{WD_j}, y, z_1, z_2, M_4, t_4\}$ to MT_i.
(5) MT_i checks the validity of t_4, calculates $\varphi = y \oplus h(ID_i \parallel k_i \parallel t_4)$, computes $ID_{WD_j} = z_1 \oplus \varphi$, $\beta = z_2 \oplus \varphi$, and $sk_{CS-MT_i} = h(TID_i \parallel ID_i \parallel k_i \parallel t_4 \parallel \varphi)$, then verifies $M_4 = h(TID_i \parallel ID_i \parallel TID_{WD_j} \parallel ID_{WD_j} \parallel \beta \parallel sk_{CS-MT_i} \parallel k_i \parallel t_4 \parallel \varphi)$. Using the r obtained during login, it calculates the pseudonym PID_{WD_j} of WD_j, selects a random number ρ and a timestamp t_5, computes $t = \rho \oplus h(PID_{WD_j} \beta \parallel t_5)$, $r = ID_i \oplus \rho$, $sk_{MT_i-WD_j} = h(PID_{WD_j} \parallel TID_i \parallel ID_i \parallel TID_{WD_j} \parallel ID_{WD_j} \parallel \rho \parallel \beta \parallel t_5)$, and $M_5 = h(PID_{WD_j} \parallel TID_i \parallel ID_i \parallel TID_{WD_j} \parallel ID_{WD_j} \parallel \rho \parallel \beta \parallel t_5 \parallel sk_{MT_i-WD_j})$, then sends $\{TID_i, t, r, t_5, M_5\}$ to WD_j.
(6) WD_j checks the validity of t_5, calculates $\rho = t \oplus h(PID_{WD_j} \parallel \beta \parallel t_5)$ and $ID_i = r \oplus \rho$, computes $sk_{MT_i-WD_j} = h(PID_{WD_j} \parallel TID_i \parallel ID_i \parallel TID_{WD_j} \parallel ID_{WD_j} \parallel \rho \parallel \beta \parallel t_5)$, and verifies $M_5 = h(PID_{WD_j} \parallel TID_i \parallel ID_i \parallel TID_{WD_j} \parallel ID_{WD_j} \parallel \rho \parallel \beta \parallel t_5 \parallel sk_{MT_i-WD_j})$. If the verification succeeds, WD_j and MT_i successfully obtain the session key.

Fig. 2 displays the Authentication and Key Agreement.

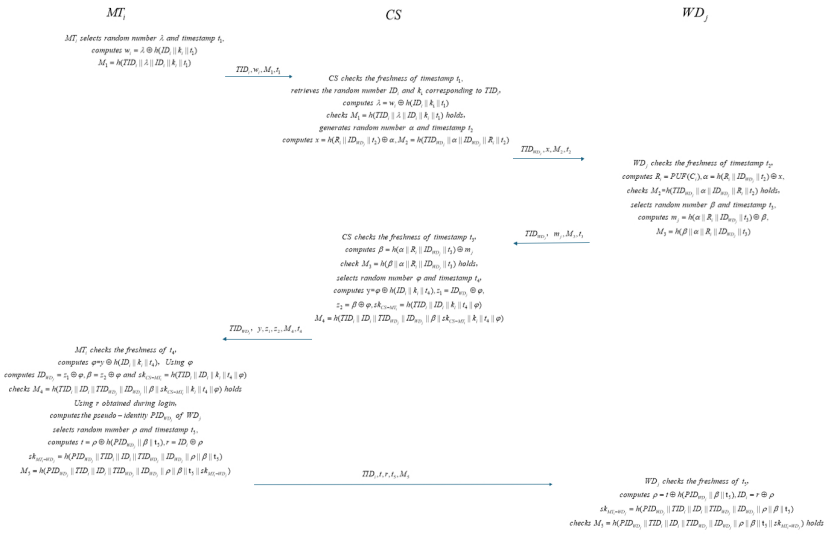

Fig. 2. Authentication and Key Agreement

6 SECURITY ANALYSIS OF PROPOSED PROTOCOL

This chapter analyzes the security of the scheme and evaluates its resistance to known attacks from adversary $\mathcal{A}$.We employ the Real-Or-Random (ROR) model to formally prove the protocol's semantic security. Assuming that the initial registration is conducted over a secure channel, this chapter concentrates on the security of communication during the authentication and key agreement phase.

This protocol involves three entities: wearable device WD_j, mobile terminal MT_i, and trusted cloud server CS. The adversary creates several instances per participant in the ROR model; we define session instances WD_j^u, MT_i^v, and CS^w for each entity, which act as oracles to process interactions and return corresponding states.we define the key concepts of the protocol and describe the queries that the adversary can perform.

Pairing Two instances I^{t_1} and I^{t_2} are considered paired if and only if both are in the "accepted" state, possess the same session identifier, and are mutually paired.

Freshness A instance I^t is considered fresh if the adversary has not obtained its session key.

Adversary It is a polynomial-time entity with full control over the communication network and can execute the following queries:
1. $Executes(WD_j^u, MT_i^v, CS^w)$: Simulates passive attacks, where $\mathcal{A}$ can eavesdrop on the communication channel via this query and output the messages exchanged between WD_j^u, MT_i^v, and CS^w.
2. $Send(I^t, m)$: Simulates active attacks, allowing $\mathcal{A}$ to send message m to the target instance I^t and obtain the instance's response.
3. $Rev(I^t)$: The adversary obtains the session key shared only between instance I^t and its communication partner via this query.
4. $CorruptWD_j(WD_j^u)$: Simulates device tampering attacks, where the adversary can intercept all secret information from the wearable device via this query.
5. $CorruptMT_i(MT_i^v)$: Simulates device tampering attacks, where the adversary can intercept all secret information stored in the mobile phone via this query.
6. $Test(I^t)$: The test checks the session key's semantic security by randomly selecting a bit b. If the session key is fresh, return the real key for $b = 1$, or a random value for $b = 0$; otherwise, return a null value $\perp$.

Semantic Security In the ROR model, semantic security means the session key is indistinguishable from a random number. The adversary initiates queries to instances and guesses a bit b' after the $Test$ query: if $b' = b$, It is said that $\mathcal{A}$ has broken the semantic security of the protocol.

We use $Succ(\mathcal{A})$ to represent the event that $\mathcal{A}$ succeeds. The advantage probability that $\mathcal{A}$ breaks semantic security is:

$$\mathrm{Adv}(\mathcal{A}) = |2\Pr[Succ(\mathcal{A})] - 1| = |2 \cdot \Pr[b = b'] - 1| \tag{1}$$

If $\mathrm{Adv}(\mathcal{A})$ is negligible, the protocol achieves semantic security under ROR.

Theorem 1. *Let $\mathcal{A}$ be a polynomial-time adversary attempting to break the proposed protocol in the ROR model. Then its advantage probability satisfies:*

$$\mathrm{Adv}(\mathcal{A}) \leq 2\left(\frac{q_h^2}{2^{l_h+1}} + \frac{q_P^2}{2\,|\mathrm{PUF}|} + \frac{(q_s+q_e)^2}{2^{l_s+1}} + \frac{q_s}{|\mathrm{PUF}|} + \max\left(C' \cdot q_s^{s'}, \frac{q_s}{2^l}\right)\right) \tag{2}$$

where q_h, q_s, q_e, and q_P represent the numbers of hash queries, send queries, execute queries, and PUF queries. l_h, l_s, and $|\mathrm{PUF}|$ represent the lengths of the hash digest, random number, and PUF response, respectively.The biometric key length is 2^l, and C' and s' are defined according to Zipf's law[20].

Proof. The proof is completed via a series of games G_i $(i = 0, \cdots, n)$ between the adversary and the challenger. The adversary's advantage in each game is denoted by Adv_{G_i}.

Game G_0 It simulates the real attack of $\mathcal{A}$ on this protocol in the sense of ROR. At the game's start, $\mathcal{A}$ guesses b, as defined:

$$\mathrm{Adv}(\mathcal{A}) = |2\mathrm{Adv}_{G_0} - 1| \tag{3}$$

Game G_1 It simulates eavesdropping attacks — $\mathcal{A}$ first queries the $Executes$ oracle to intercept messages transmitted between legitimate entities, then performs a $Test$ query to verify whether the eavesdropped data is a session key or a random number. In this protocol, the messages transmitted between entities do not include the session keys $sk_{CS-MT_i} = h(TID_i \parallel ID_i \parallel k_i \parallel t_4 \parallel \varphi)$ and $sk_{MT_i-WD_j} = h(PID_{WD_j} \parallel TID_i \parallel ID_i \parallel TID_{WD_j} \parallel ID_{WD_j} \parallel \rho \parallel \beta \parallel t_5)$

The required parameters $< ID_i, k_i, \varphi >$ and $< PID_{WD_j}, ID_i, ID_{WD_j}, \rho >$ are not included either. Therefore, the adversary cannot improve its ability to guess the session key, and the advantage of G_1 is the same as that of G_0:

$$\mathrm{Adv}_{G_1} = \mathrm{Adv}_{G_0} \tag{4}$$

Game G_2 It simulates active attacks — $\mathcal{A}$ intercepts/forges messages via $Send$ and $Hash$ queries. The protocol messages contain random numbers and hash digests, and key messages are protected by PUF responses; according to the birthday paradox, the collision probabilities of hash, PUF, and random numbers are $\frac{q_h^2}{2^{l_h+1}}$, $\frac{q_P^2}{2|\mathrm{PUF}|}$, and $\frac{(q_s+q_e)^2}{2^{l_s+1}}$, respectively. Thus, G_2 is indistinguishable from G_1, and the advantage difference satisfies:

$$|\mathrm{Adv}_{G_2} - \mathrm{Adv}_{G_1}| \leq \frac{q_h^2}{2^{l_h+1}} + \frac{q_P^2}{2\,|\mathrm{PUF}|} + \frac{(q_s+q_e)^2}{2^{l_s+1}} \tag{5}$$

Game G_3: Node Capture Attack The adversary executes the $Corruptwd_j(WD_j^u)$ query to extract all secret parameters stored in WD_j, but cannot obtain the PUF response $R_i = \mathrm{PUF}(C_i)$. The adversary invokes $Send(CS, m')$ to forge a message m' of WD_j, attempting to pass CS's verification. However, the forged message m' needs to match the R_i stored in CS to pass verification; at this point, the adversary can only guess R_i via brute-force attacks. Thus, the adversary initiates q_s $Send(CS, m')$ queries, and the advantage difference satisfies:

$$|\mathrm{Adv}_{G_3} - \mathrm{Adv}_{G_2}| \leq \frac{q_s}{|\mathrm{PUF}|} \tag{6}$$

Game G_4 The adversary executes the $Corruptmt_i(MT_i^v)$ query to extract all secret parameters stored in MT_i, and tries to compute the session key SK via these parameters. However, each parameter contains the user's password and biometric data; the adversary must guess both items simultaneously. Guessing the biometric key has a probability of $\frac{1}{2^l}$, and the password's guessing probability follows the Zipf [63] model. The advantage difference satisfies:

$$|\mathrm{Adv}_{G_4} - \mathrm{Adv}_{G_3}| \leq \max\left(C' \cdot q_s^{s'}, \frac{q_s}{2^l}\right) \tag{7}$$

All oracles have been emulated. The probability that $\mathcal{A}$ guesses b is equivalent to flipping a coin, so $\mathrm{Adv}_{G_4} = \frac{1}{2}$.

To summarize:

$$\mathrm{Adv}(\mathcal{A}) = |2\mathrm{Adv}_{G_0} - 1| = 2\,|\mathrm{Adv}_{G_0} - \mathrm{Adv}_{G_4}| \tag{8}$$

and it satisfies

$$\mathrm{Adv}(\mathcal{A}) \leq 2\left(\frac{q_h^2}{2^{l_h+1}} + \frac{q_P^2}{2\,|\mathrm{PUF}|} + \frac{(q_s+q_e)^2}{2^{l_s+1}} + \frac{q_s}{|\mathrm{PUF}|} + \max\left(C' \cdot q_s^{s'}, \frac{q_s}{2^l}\right)\right). \tag{9}$$

Therefore, Theorem 1 proves the session key security of the proposed scheme.

7 PERFORMANCE ANALYSIS

This chapter evaluates the computational and communication overheads of the proposed scheme, comparing it with the schemes by Sahoo et al[16] and Li et al[13].Relevant experiments were performed on a Windows 10 laptop equipped with an AMD Ryzen 5 4500U processor (2.38 GHz), Radeon graphics, and 8GB of RAM. We use python standard library and third-party libraries to perform calculations and simulate the execution time of the main cryptographic operations involved in the proposed protocol, where the execution times are as follows: hash operation T_h =0.0018 ms, elliptic curve point multiplication T_p =1.031 ms, symmetric encryption or decryption T_s =0.375 ms, and fuzzy extractor T_{fe} =1.031 ms.

7.1 Computational Overhead

This subsection evaluates the computational overhead of the proposed protocol (covering user login, authentication, and key agreement) in comparison with baseline protocols. with the execution times of the relevant cryptographic primitives provided earlier; operations with minimal computational cost (e.g., data XOR and concatenation) are not included in the overhead evaluation scope.

The computational overhead of each protocol is shown in Table 2. For the proposed protocol, the computational overhead required by the wearable device is $7T_h$, which is lower than the schemes by Sahoo et al[16]. and Li et al[13], proving that our scheme is better suited for resource-limited wearable devices. The total computational overhead of our protocol is $29T_h + T_{fe}$. Compared with the other two schemes, our protocol maintains low computational overhead while ensuring superior security.

Table 2. Comparison of computation costs

Scheme	Computation cost of WD_j (ms)	Total computation cost (ms)
Sahoo et al[16]	$8T_h + 2T_p + 2T_s$=2.8264	$28T_h + 5T_p + 6T_s$=7.4554
Li et al[13]	$5T_h + 2T_s$=0.759	$18T_h + 10T_s$=3.7824
Ours	$7T_h$=0.0126	$29T_h + T_{fe}$=1.0832

7.2 Communication Overhead

Table 3 compares the communication overhead of the protocols by Sahoo et al[16], Li et al[13], and our proposed protocol. We set the bit lengths of relevant parameters as follows: 128 bits for identity identifiers and random numbers, 320 bits for elliptic curve cryptography (ECC) points, 128 bits for the hash algorithm output length, and 32 bits for timestamps.

Regarding the communication overhead of each protocol: the wearable device communication overhead of Sahoo et al.'s protocol[16] is 736, with a total communication overhead of 2944; the wearable device communication overhead of Li et al.'s protocol[13] is 576, with a total communication overhead of 2912; the wearable device communication overhead of our protocol is 416, with a total communication overhead of 2464.

The communication overhead of our protocol is less than that of Sahoo et al[16] and Li et al[13], making it more suitable for resource-limited environments.

8 Conclusion

This paper proposes a lightweight anonymous authentication protocol that combines PUF and fuzzy extractor, aiming to address secure authentication and

Table 3. Comparison of communication costs

Scheme	Communication cost of WD_j	Total communication cost
Sahoo et al[16]	736 (bits)	2944 (bits)
Li et al[13]	576 (bits)	2912 (bits)
Ours	416 (bits)	2464 (bits)

key agreement in the three-party collaborative environment of wearable devices. The scheme leverages PUF's inherent uniqueness and unclonability, reducing the computational and storage overheads while maintaining security. Additionally, the protocol structure avoids the issue of cloud servers obtaining the session key between devices and terminals, enhancing the system's end-to-end security. Security analysis within the ROR model proves the protocol's security against attacks such as session key leakage and device capture. Experimental results demonstrate superior performance in computational and communication overheads compared to existing schemes, making the protocol highly suitable for resource-limited wearable devices and providing a practical solution for secure, efficient wearable system designs.

Acknowledgments. The work was supported by the Natural Science Foundation of Zhejiang Province (No.LQN25F020006).

References

1. Ai, X., Badshah, A., Tu, S., Waqas, M., Ahmad, I.: An improved ultra-lightweight anonymous authenticated key agreement protocol for wearable devices. IEEE Transactions on Mobile Computing (2025)
2. Alruwaili, O., Tanveer, M., Alotaibi, F.M., Abdelfattah, W., Armghan, A., Alserhani, F.M.: Securing the iot-enabled smart healthcare system: A puf-based resource-efficient authentication mechanism. Heliyon **10**(18) (2024)
3. Byun, W., Je, M., Kim, J.H.: Advances in wearable brain-computer interfaces from an algorithm-hardware co-design perspective. IEEE Transactions on Circuits and Systems II: Express Briefs **69**(7), 3071–3077 (2022)
4. Challa, S., Das, A.K., Odelu, V., Kumar, N., Kumari, S., Khan, M.K., Vasilakos, A.V.: An efficient ecc-based provably secure three-factor user authentication and key agreement protocol for wireless healthcare sensor networks. Computers & Electrical Engineering **69**, 534–554 (2018)
5. Chen, C.M., Li, Z., Chaudhry, S.A., Li, L.: Attacks and solutions for a two-factor authentication protocol for wireless body area networks. Security and Communication Networks **2021**(1), 3116593 (2021)
6. Dodis, Y., Reyzin, L., Smith, A.: Fuzzy extractors: How to generate strong keys from biometrics and other noisy data. In: International conference on the theory and applications of cryptographic techniques. pp. 523–540. Springer (2004)
7. Dolev, D., Yao, A.: On the security of public key protocols. IEEE Transactions on information theory **29**(2), 198–208 (2003)
8. Fotouhi, M., Bayat, M., Das, A.K., Far, H.A.N., Pournaghi, S.M., Doostari, M.A.: A lightweight and secure two-factor authentication scheme for wireless body area networks in health-care iot. Computer Networks **177**, 107333 (2020)

9. Guo, Y., Zhang, Z., Guo, Y.: Anonymous authenticated key agreement and group proof protocol for wearable computing. IEEE Transactions on Mobile Computing **21**(8), 2718–2731 (2021)
10. cheol Jeong, I., Bychkov, D., Searson, P.C.: Wearable devices for precision medicine and health state monitoring. IEEE Transactions on Biomedical Engineering **66**(5), 1242–1258 (2018)
11. Jiang, Q., Ma, J., Yang, C., Ma, X., Shen, J., Chaudhry, S.A.: Efficient end-to-end authentication protocol for wearable health monitoring systems. Computers & Electrical Engineering **63**, 182–195 (2017)
12. Lee, C., Oh, M., Kwon, D., Park, Y., Park, Y.: Plaka-md: Puf-based lightweight authentication and key agreement scheme for medical devices in ioht. IEEE Internet of Things Journal (2025)
13. Li, X., Niu, J., Kumari, S., Liao, J., Liang, W., Khan, M.K.: A new authentication protocol for healthcare applications using wireless medical sensor networks with user anonymity. Security and Communication Networks **9**(15), 2643–2655 (2016)
14. Liu, W., Liu, H., Wan, Y., Kong, H., Ning, H.: The yoking-proof-based authentication protocol for cloud-assisted wearable devices. Personal and Ubiquitous Computing **20**(3), 469–479 (2016)
15. Rawat, G.S., Singh, K., Arshad, N.I., Hadidi, K., Ahmadian, A.: A lightweight authentication scheme with privacy preservation for vehicular networks. Computers and Electrical Engineering **100**, 108016 (2022)
16. Sahoo, S.S., Mohanty, S., Sahoo, K.S., Daneshmand, M., Gandomi, A.H.: A three-factor-based authentication scheme of 5g wireless sensor networks for iot system. IEEE internet of things journal **10**(17), 15087–15099 (2023)
17. Suh, G.E., Devadas, S.: Physical unclonable functions for device authentication and secret key generation. In: Proceedings of the 44th annual design automation conference. pp. 9–14 (2007)
18. Tanveer, M., Ahmad, M., Nguyen, T.N., Abd El-Latif, A.A., et al.: Resource-efficient authenticated data sharing mechanism for smart wearable systems. IEEE Transactions on Network Science and Engineering **10**(5), 2525–2536 (2022)
19. Tu, S., Badshah, A., Alasmary, H., Waqas, M.: Eake-wc: Efficient and anonymous authenticated key exchange scheme for wearable computing. IEEE Transactions on Mobile Computing **23**(5), 4752–4763 (2023)
20. Wang, D., Cheng, H., Wang, P., Huang, X., Jian, G.: Zipf's law in passwords. IEEE Transactions on Information Forensics and Security **12**(11), 2776–2791 (2017)
21. Wang, S., Bie, R., Zhao, F., Zhang, N., Cheng, X., Choi, H.A.: Security in wearable communications. IEEE Network **30**(5), 61–67 (2016)
22. Wang, W., Chen, Q., Yin, Z., Srivastava, G., Gadekallu, T.R., Alsolami, F., Su, C.: Blockchain and puf-based lightweight authentication protocol for wireless medical sensor networks. IEEE Internet of Things Journal **9**(11), 8883–8891 (2021)
23. Wu, T.Y., Wang, L., Chen, C.M.: Enhancing the security: A lightweight authentication and key agreement protocol for smart medical services in the ioht. Mathematics **11**(17), 3701 (2023)
24. Zheng, Y.L., Ding, X.R., Poon, C.C.Y., Lo, B.P.L., Zhang, H., Zhou, X.L., Yang, G.Z., Zhao, N., Zhang, Y.T.: Unobtrusive sensing and wearable devices for health informatics. IEEE transactions on biomedical engineering **61**(5), 1538–1554 (2014)
25. Zhou, J., Cao, Z., Dong, X., Lin, X.: Security and privacy in cloud-assisted wireless wearable communications: Challenges, solutions, and future directions. IEEE wireless Communications **22**(2), 136–144 (2015)