A Provably Secure Lightweight Key Agreement Protocol for Wireless Body Area Networks in Healthcare System

Maryam Zia, Mohammad S. Obaidat, *Life Fellow, IEEE*, Khalid Mahmood, Salman Shamshad, Muhammad Asad Saleem, and Shehzad Ashraf Chaudhry

***Abstract*—Wireless Body Area Network (WBAN) is a vital application of the Internet of Things (IoT) that plays a significant role in gathering a patient's healthcare information. This collected data helps special professionals like doctors or physicians analyze patients' health status to cure different diseases. However, collecting such information from an insecure channel can be threatening due to the potential security threats. Therefore, it is crucial to secure this sensitive information. This article proposes a secure and lightweight authentication protocol for WBAN. The devised protocol is scalable, secure, and lightweight compared to various relevant competing protocols. The informal security analysis shows that the designed protocol is lightweight, secure, and efficient in resisting various major attacks. The performance analysis demonstrates our protocol's supremacy over various competing protocols in terms of computation and communication costs, inducing efficiency of 20.3% and 12.3%, respectively. Moreover, the practical performance of the designed protocol from the network point of view is measured using the widely recognized NS3 simulation tool.**

Manuscript received 4 January 2022; revised 19 March 2022 and 14 June 2022; accepted 26 August 2022. Date of publication 30 August 2022; date of current version 13 December 2022. Paper no. TII-22-0009. *(Corresponding author: Khalid Mahmood.)*

Maryam Zia and Muhammad Asad Saleem are with the Department of Computer Science, COMSATS University Islamabad, Sahiwal 57000, Pakistan (e-mail: maryamzia935@gmail.com; masadsaleem123@gmail.com).

Mohammad S. Obaidat is with the Department of Computer Science, and Cybersecurity Center, University of Texas-Permian Basin, Odessa, TX 79762 USA, also with the King Abdullah II School of Information Technology, University of Jordan, Amman 11942, Jordan, also with the School of Computer and Communication Engineering, University of Science and Technology Beijing, Beijing 100083, China, and also with the Amity University, Noida, UP 201301, India (e-mail: msobaidat@gmail.com).

Khalid Mahmood is with the Future Technology Research Center, National Yunlin University of Science and Technology, Yunlin 64002, Taiwan (e-mail: khalidm.research@gmail.com).

Salman Shamshad is with the Department of Software Engineering, The University of Lahore, Lahore 54590, Pakistan (e-mail: salmanshamshad01@gmail.com).

Shehzad Ashraf Chaudhry is with the Department of Computer Science and Information Technology, College of Engineering, Abu Dhabi University, Abu Dhabi, UAE, and also with the Department of Computer Engineering, Faculty of Engineering and Architecture, Nisantasi University, 34398 Istanbul, Turkey (e-mail: ashraf.shehzad.ch@gmail.com).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TII.2022.3202968.

Digital Object Identifier 10.1109/TII.2022.3202968

***Index Terms*—Authentication, authentication protocol, Internet of Things (IoT), mutual authentication, security, wireless body area networks (WBANs).**

I. Introduction

THE Internet of Things (IoT) technology includes machine-to-machine communication that can share all information via the Internet without human interaction. IoT consists of different interconnected computing devices. These devices play a vital role in obtaining information to bring efficiency and comfort to our lives. These devices include smart glasses, health trackers, smart watches, smart uniforms, wristbands, on-body cameras, etc. IoT also plays a vital role in the healthcare system; these days, human life depends on e-healthcare services s [1]. IoT usually combines Artificial Intelligence (AI) systems to monitor the patient's health.

The Wireless Body Area Network (WBAN) is a crucial application of IoT that plays an essential role in smart healthcare services to gather the patient's healthcare information. WBANs provide healthcare services to critical patients at hospitals or homes [2]. In 1996, this technology was introduced by Zimmerman [3]. WBAN provides a facility for doctors to check the patient's health status remotely through short-distance wireless communication platforms such as Wi-Fi. Wearable devices are fixed to the patient's body to sense and analyze secret and delicate information [4], [5].

A WBAN system consists of three main modules such as mid node ($\mathcal{MN}$), wearable node ($\mathcal{WN}$), and controller node ($\mathcal{CN}$). $\mathcal{CN}$ gathers the entire patient's health data from $\mathcal{WN}$ through $\mathcal{MN}$. This architecture consists of three main tiers, which are displayed in Fig. 1. Tier 1 displays the connection between the $\mathcal{WN}$ and $\mathcal{MN}$. $\mathcal{WN}$ is attached to the patient's body, like wearable blood pressure monitors, glucose levels, electroencephalogram, oxygen saturation, and electrocardiography. $\mathcal{WN}$ senses, analyzes, stores, and transmits secret and delicate information to a central node (mid node) via a close-range wireless medium such as Wi-Fi or Bluetooth. Tier 2 displays the connection between $\mathcal{MN}$ and $\mathcal{CN}$. $\mathcal{MN}$ forwards the collected information from the sensor/wearable nodes to $\mathcal{CN}$. Tier 3 attaches $\mathcal{CN}$ or local server to the patient's health data

TABLE I
EXISTING AUTHENTICATION PROTOCOLS: A COMPARATIVE STUDY

Protocols	Year	Cryptographic primitives	Advantages/descriptions	Drawbacks/limitations
Chen et al. [7]	2018	∗ Hash function	∗ Provide user anonymity ∗ Provide security against impersonation attacks	∗ Unsuitable for real life applications ∗ Vulnerable to intermediate node capture attack
Koya and Deepthi [8]	2018	∗ BAN logic ∗ One-way hash method	∗ Solved key escrow issue ∗ Provides mutual authentication and key agreement	∗ Required high computational and high cost ∗ Problem is to gathering and converting the physiological inputs on all sensors
Kompara et al. [9]	2019	∗ BAN logic ∗ Hash function ∗ XOR operation	∗ Mutual authentication ∗ Reduce communication cost ∗ Efficient ∗ Session key establishment	∗ Retaining computational complexity ∗ Limited processing power ∗ Limited memory and energy ∗ More communication cost
Sowjanya et al. [10]	2020	∗ Elliptic curve cryptography (ECC) ∗ Cryptographic hash function ∗ BAN logic	∗ Removes the security loopholes ∗ Mutual authentication ∗ Reduces the overall complexity ∗ Session key establishment	∗ Higher computation ∗ Resource constrained ∗ No lightweight three factor authentication protocol ∗ Less provide security and privacy
Gupta et al. [11]	2020	∗ XOR operation ∗ Hash function	∗ Safe from replay attack ∗ Suitable for real-life applications ∗ Reduce complexity	∗ Incorrectness ∗ Sensor node physical capture attack ∗ Sensor node anonymity violation attack
Hussain et al. [12]	2021	∗ Hash function	∗ Untraceability of application provider and client ∗ Safe from desynchronization attack ∗ Prevent from forgery attack	∗ Replay attack ∗ Vulnerable to impersonation attack ∗ Communication overhead

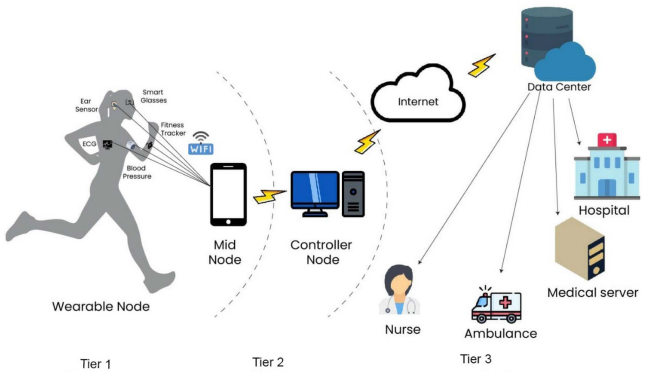

Fig. 1. Architecture of WBAN.

center through the Internet. The local server transmits all the collected data to the health data center. These collected data help special professionals like doctors or physicians to analyze patients' health status to treat different diseases. On the other hand, patients may save time and money by utilizing numerous health services at home [6].

The WBAN platform carries and exchanges sensitive critical patient information that needs to be secured against potential security threats as the patient data roams through public channels. Different protocols have been proposed to enhance secure user authentication, improving the perfect forward secrecy, privacy, anonymity, etc. Numerous researchers have proposed secure mutual authentication and key exchange protocols for various contexts such as wireless sensor networks, multiserver, and single server environments. Recently, several researchers have started more research on WBAN to protect the different security key challenges and enhance overall efficiency.

After critically evaluating the facts in Table I, we can conclude that different authentication protocols [7], [8], [9], [10], [11], [12] have been proposed for wearable nodes to achieve the secure transmission of health data. Nonetheless, most of the existing authentication protocols are inadequate to secure intermediate node physical capture attacks, untraceability and lack of anonymity, impersonation attacks, etc. Besides, many authentication protocols are not lightweight due to their expensive communication and computation overhead. Hence, it becomes a necessity of time to design security and privacy solution for WBAN. Therefore, we proposed a lightweight and secure mutual authentication protocols for WBANs.

The major contributions of the proposed protocol are as follows.

1) First, we analyze the security of competing protocols and show that they are vulnerable to various threats.
2) We then proposed a key agreement protocol that is lightweight, secure, and efficient in nature.
3) We present a formal and informal security analysis of our introduced protocol to demonstrate the security strength of the designed protocol.
4) The performance analysis demonstrates that our designed protocol is better than other relevant protocols.
5) We have also illustrated the network performance of our designed protocol through a globally recognized NS3 simulator tool.

A. Article Organization

The rest of this article is sorted as follows. Some cryptographic preliminaries are presented in Section II. The proposed protocol is discussed in Section III. Security analysis is given in Section IV and NS-3 simulation results are discussed in Section V. Performance evaluation of the proposed protocol is presented in Section VI. At last, the entire work is summarized in Section VII.

II. PRELIMINARIES

This section introduces the network and threat models to describe the work of this article. Whereas, the cryptographic symbols are summarized in Table II.

A. Network Model

The organized WBAN network model of our protocol is depicted in Fig. 2. It consists of the following four different types of nodes:

1) Wearable Node ($\mathcal{WN}$);
2) Mid Node ($\mathcal{MN}$);
3) Controller Node ($\mathcal{CN}$);

TABLE II
SYMBOLS AND THEIR DESCRIPTION

Symbols	Description	Symbols	Description
$\mathcal{A}_{\mathcal{K}}$	Adversary	$\mathcal{WN}$	Wearable node
id_w	Identity of a wearable node	tid_w	Temporary identity of a wearable node
$\mathrm{tid}_w^{\mathrm{new}}$	New temporary identity of wearable node	$\mathcal{MN}$	Mid node
$\mathrm{id}_{\mathrm{mn}}$	Identity of mid node	$\mathcal{CN}$	Controller node
$\mathcal{K}_{\mathrm{cn}}$	Secret key of $\mathcal{CN}$	$\mathcal{E}_{\mathcal{K}_{\mathrm{cn}}}$	Encryption using $\mathcal{CN}$ secret key
$\mathcal{D}_{\mathcal{K}_{\mathrm{cn}}}$	Decryption using $\mathcal{CN}$ secret key	$\mathcal{SA}$	System administrator
$\mathcal{L}_{\mathrm{sk}}$	Long term shared secret key	$\mathcal{T}_i$	Timestamp
$\Delta\mathcal{T}_m$	Maximum transmission delay	$\|$	Concatenation operator
$\oplus$	XOR operator	$\mathcal{S}_{\mathcal{K}}$	Session key between $\mathcal{WN}$ and $\mathcal{CN}$
$h(.)$	Cryptography hash function		

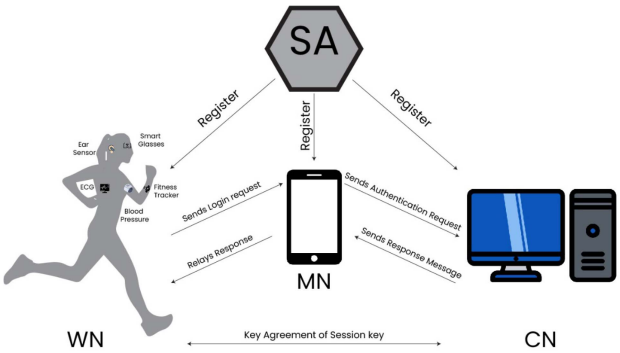

Fig. 2. Network model of the proposed protocol.

4) System Administrator ($\mathcal{SA}$).

Primarily, $\mathcal{SA}$ is responsible to register $\mathcal{WN}$, and $\mathcal{MN}$ into the WBAN network. $\mathcal{WN}$ is assumed to be a resource-limited wearable/sensor node that monitor a patient's real-time healthcare information. $\mathcal{MN}$ has more processing and communication capacity than $\mathcal{WN}$'s, and it is responsible for exchanging the authentication messages between $\mathcal{WN}$ and $\mathcal{WN}$. $\mathcal{CN}$ is the backbone of the whole WBAN network that verifies the authenticity of $\mathcal{MN}$ and $\mathcal{WN}$ before the establishment of session key. After successfully passing the verification check, $\mathcal{WN}$ can establish a session key with $\mathcal{CN}$ for secure communication.

B. Threat Model

We used the known Dolev–Yao ($\mathcal{DY}$) [13] and Canetti–Krawczyk ($\mathcal{CK}$) [14] threat models, where all entities communicate and send messages through a public channel. According to [15], [16], and [17], we summarize attacker $\mathcal{A}_{\mathcal{K}}$ capabilities in this article as follows.

1) An $\mathcal{A}_{\mathcal{K}}$ has the ability to handle all public channel communication and can replay, block, remove, alter, and interpret messages.
2) The communication between $\mathcal{WN}$, $\mathcal{MN}$, and $\mathcal{CN}$ is considered public, and $\mathcal{A}_{\mathcal{K}}$ has access to the public channel's messages.
3) An $\mathcal{A}_{\mathcal{K}}$ can retrieve all the values from a physically captured $\mathcal{MN}$'s memory using a power analysis attack.
4) $\mathcal{A}_{\mathcal{K}}$ may be capable of physically capturing different numbers of $\mathcal{WN}$s and retrieving all the information in them using a power analysis attack.
5) Users and sensors cannot be trusted, and any user can impersonate another legal user.
6) Secret parameters of involving entities are secured, and $\mathcal{A}_{\mathcal{K}}$ cannot reveal them.

III. PROPOSED PROTOCOL

This section describes our protocol for WBAN.

A. Initialization Phase

The first phase of our designed protocol is the system initialization. This phase is accomplished by the system administrator ($\mathcal{SA}$) in an offline mode. The controller node ($\mathcal{CN}$) is enrolled by the $\mathcal{SA}$, and the steps are given below.

IP 1: $\mathcal{SA}$ chooses a $\mathcal{K}_{\mathrm{cn}}$ as a master key for $\mathcal{CN}$
IP 2: Securely stores $\mathcal{K}_{\mathrm{cn}}$ it in $\mathcal{CN}'s$ memory.
IP 3: Next, the $\mathcal{SA}$ selects a "one-way hash function" which takes an arbitrarily bit-length input string and generates a fixed-length output as "message digest (hash value)". Hash value and $\mathcal{CN}$ secret key consume 256 b.

B. Registration Phase

This subsection describes that $\mathcal{SA}$ initiates the registration phase. The following subsequent steps are executed among $\mathcal{WN}$, $\mathcal{MN}$, $\mathcal{CN}$, and $\mathcal{SA}$.

Step 1: $\mathcal{SA}$ selects an id_w as a secret identity for $\mathcal{WN}$. $\mathcal{SA}$ also chooses a unique long-term secret shared key $\mathcal{L}_{\mathrm{sk}}$ for $\mathcal{MN}$ and $\mathcal{WN}$.

Step 2: It then calculates $a_m \longleftarrow \mathrm{id}_w \oplus h(\mathrm{id}_w \parallel \mathcal{K}_{\mathrm{cn}})$, and $c_m \longleftarrow h(\mathcal{K}_{\mathrm{cn}} \parallel \mathrm{id}_w)$. Afterwards, it selects r_{sa} and computes $\mathrm{tid}_w \longleftarrow \mathcal{E}_{K_{\mathrm{cn}}}(\mathrm{id}_w \parallel r_{\mathrm{sa}})$.

Step 3: $\mathcal{SA}$ also selects an $\mathrm{id}_{\mathrm{mn}}$ unique identity of $\mathcal{MN}$ and computes $X_{\mathrm{in}} \longleftarrow h(\mathrm{id}_{\mathrm{mn}} \parallel \mathcal{K}_{\mathrm{cn}})$ for $\mathcal{MN}$.

Step 4: Finally, it stores $\{L_{\mathrm{sk}}, a_m, c_m, \mathrm{tid}_w\}$ in $\mathcal{WN}'s$ memory and $\{\mathrm{id}_{\mathrm{mn}}, \mathcal{L}_{\mathrm{sk}}, \mathcal{X}_{\mathrm{in}}\}$ in $\mathcal{MN}'s$ memory.

Step 5: It also maintains the unique identity $\mathrm{id}_{\mathrm{mn}}$ of $\mathcal{MN}$ in the memory of $\mathcal{CN}$.

C. Key Agreement Phase

This subsection presents the key agreement phase of the designed protocol. During the agreement phase, $\mathcal{WN}$ undergoes a mutual authentication phase with $\mathcal{CN}$ via $\mathcal{MN}$. The subsequent steps are performed among $\mathcal{WN}$, $\mathcal{CN}$, and $\mathcal{MN}$:

Step 1: $\mathcal{WN}$ chooses random number r_m and calculates $x_m \longleftarrow a_m \oplus \mathrm{id}_w$ and $y_m \longleftarrow x_m \oplus r_m$. It also generates current timestamp $\mathcal{T}_1$ and calculate $W_n \longleftarrow h(\mathrm{id}_w \parallel \mathrm{tid}_w \parallel c_m \parallel r_m \parallel \mathcal{T}_1)$ and sends $\{\mathrm{tid}_w, y_m, a_m, W_n, \mathcal{T}_1\}$ to $\mathcal{MN}$ via a public channel.

Step 2: After receiving this login request message, $\mathcal{MN}$ first verifies the validity of timestamp, i.e., $|\mathcal{T}_1 - \mathcal{T}_c| < \Delta\mathcal{T}_m$ where T_c is $\mathcal{MN}'s$ current timestamp. It generates current timestamp $\mathcal{T}_2$ to calculate $V_m \longleftarrow h(\mathrm{id}_{\mathrm{mn}} \parallel X_{in} \parallel \mathcal{T}_2)$ and sends $\{\mathrm{tid}_w, y_m, a_m, V_m, \mathrm{id}_{\mathrm{mn}}, \mathcal{T}_1, \mathcal{T}_2\}$to $\mathcal{CN}$ through a public channel.

Step 3: After receiving this login request message, $\mathcal{CN}$ first verifies the validity of timestamp, i.e., $|\mathcal{T}_2 - \mathcal{T}_c| < \Delta\mathcal{T}_m$ or not, where T_c is $\mathcal{CN}$ current timestamp and

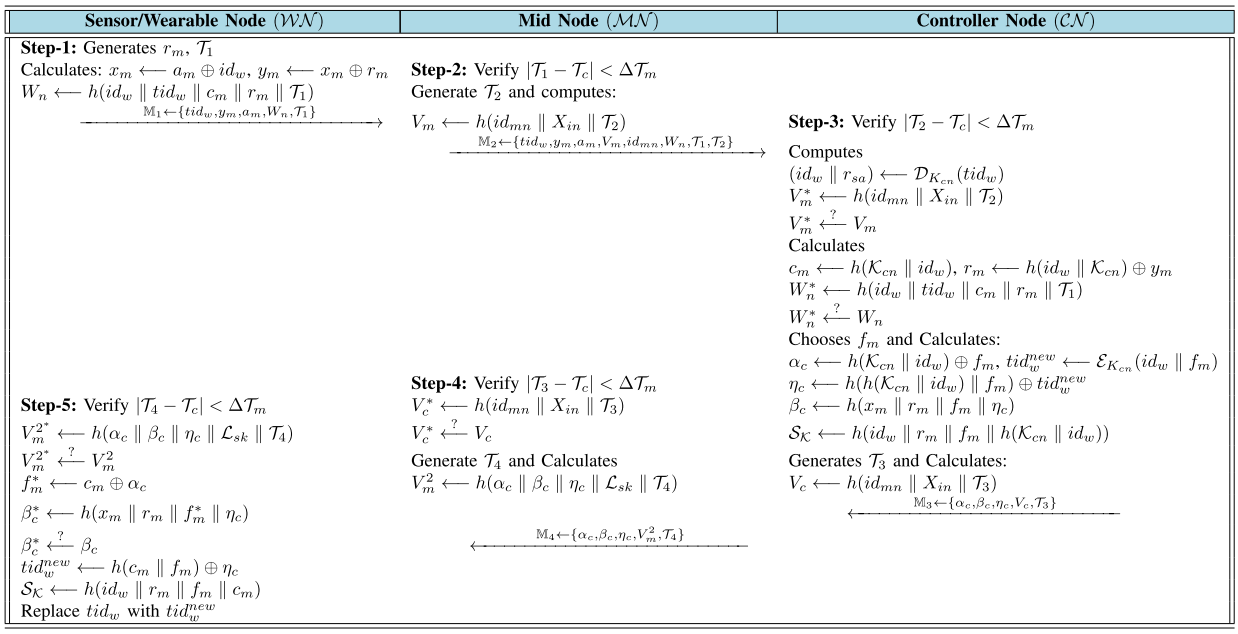

Fig. 3. Key agreement phase of proposed protocol.

then computes $(\mathrm{id}_w \| r_{sa}) \longleftarrow \mathcal{D}_{K_{\mathrm{cn}}}(\mathrm{tid}_w)$, $V_m^* \longleftarrow h(\mathrm{id}_{\mathrm{mn}} \| X_{in} \| \mathcal{T}_2)$ and verifies whether $V_m^* \stackrel{?}{\longleftarrow} V_m$ or not. If the verification fails, $\mathcal{CN}$ terminates the connection. Otherwise, it then calculates $c_m \longleftarrow h(\mathcal{K}_{\mathrm{cn}} \| \mathrm{id}_w)$, $r_m \longleftarrow h(\mathrm{id}_w \| \mathcal{K}_{\mathrm{cn}}) \oplus y_m$, $W_n^* \longleftarrow h(\mathrm{id}_w \| \mathrm{tid}_w \| c_m \| r_m \| \mathcal{T}_1)$, and verifies $W_n^* \stackrel{?}{\longleftarrow} W_n$ or not. If the verification fails, $\mathcal{CN}$ terminates the connection. Otherwise, it chooses f_m and further calculates $\alpha_c \longleftarrow h(\mathcal{K}_{\mathrm{cn}} \| \mathrm{id}_w) \oplus f_m$, $\mathrm{tid}_w^{\mathrm{new}} \longleftarrow \mathcal{E}_{K_{\mathrm{cn}}}(\mathrm{id}_w \| f_m)$, $\eta_c \longleftarrow h(h(\mathcal{K}_{\mathrm{cn}} \| \mathrm{id}_w) \| f_m) \oplus \mathrm{tid}_w^{\mathrm{new}}$, $\beta_c \longleftarrow h(x_m \| r_m \| f_m \| \eta_c)$ and $\mathcal{S}_\mathcal{K} \longleftarrow h(\mathrm{id}_w \| r_m \| f_m \| h(\mathcal{K}_{\mathrm{cn}} \| \mathrm{id}_w))$. Furthermore, $\mathcal{CN}$ generates the current time stamp $\mathcal{T}_3$ and calculate $V_c \longleftarrow h(\mathrm{id}_{\mathrm{mn}} \| X_{in} \| \mathcal{T}_3)$. $\mathcal{CN}$ sends $\{\alpha_c, \beta_c, \eta_c, V_c, \mathcal{T}_3\}$ to $\mathcal{MN}$ through a public channel.

Step 4: After receiving this message, $\mathcal{MN}$ first verifies the validity of timestamp, i.e., $|\mathcal{T}_3 - \mathcal{T}_c| < \Delta\mathcal{T}_m$ or not. Then, further calculates $V_c^* \longleftarrow h(\mathrm{id}_{\mathrm{mn}} \| X_{in} \| \mathcal{T}_3))$ and verifies $V_c^* \stackrel{?}{\longleftarrow} V_c$ or not. If the verification fails, $\mathcal{MN}$ terminates the connection. Otherwise, it generates the time stamp $\mathcal{T}_4$ and then calculates $V_m^2 \longleftarrow h(\alpha_c \| \beta_c \| \eta_c \| \mathcal{L}_{\mathrm{sk}} \| \mathcal{T}_4)$. Thereafter, $\mathcal{MN}$ sends response message, i.e., $\{\alpha_c, \beta_c, \eta_c, V_m^2, \mathcal{T}_4\}$ to $\mathcal{WN}$.

Step 5: After receiving the message, $\mathcal{WN}$ first verifies the validity of timestamp, i.e., $|\mathcal{T}_4 - \mathcal{T}_c| < \Delta\mathcal{T}_m$ or not. Then, $\mathcal{WN}$ further calculates $V_m^{2^*} \longleftarrow h(\alpha_c \| \beta_c \| \eta_c \| \mathcal{L}_{\mathrm{sk}} \| \mathcal{T}_4)$. Verify $V_m^{2^*} \stackrel{?}{\longleftarrow} V_m^2$. If the verification fails, $\mathcal{WN}$ terminates the connection. Otherwise, $\mathcal{WN}$ calculates $f_m^* \longleftarrow c_m \oplus \alpha_c$, $\beta_c^* \longleftarrow h(x_m \| r_m \| f_m^* \| \eta_c)$ and verifies $\beta_c^* \stackrel{?}{\longleftarrow} \beta_c$. if the verification fails, $\mathcal{WN}$ terminates the connection. Otherwise, it calculates $\mathrm{tid}_w^{\mathrm{new}} \longleftarrow h(c_m \| f_m) \oplus \eta_c$, and $\mathcal{S}_\mathcal{K} \longleftarrow h(\mathrm{id}_w \| r_m \| f_m \| c_m)$. It replaces tid_w with $\mathrm{tid}_w^{\mathrm{new}}$ in its own memory.

Finally, both $\mathcal{WN}$ and $\mathcal{CN}$ agree with common exchange session key $\mathcal{S}_\mathcal{K}$ and share the session key. The key-agreement phase is also demonstrated in Fig. 3

D. Dynamic Node Updating Phase

It is possible that a new $\mathcal{WN}$ is needed to add or replace with the exisitng $\mathcal{WN}$s. However, a new $\mathcal{WN}$ must be dynamically added into the WBAN system. When a new $\mathcal{WN}$ joins an existing system say $(\mathcal{WN}^{\mathrm{new}})$, $\mathcal{SA}$ executes the following steps:

Step 1: $\mathcal{SA}$ chooses a unique identity $\mathrm{id}_w^{\mathrm{new}}$, and a long-term shared secret key $\mathcal{L}_{\mathrm{sk}}^{\mathrm{new}}$.

Step 2: It then calculates $a_m^{\mathrm{new}} \longleftarrow \mathrm{id}_w^{\mathrm{new}} \oplus h(\mathrm{id}_w^{\mathrm{new}} \| \mathcal{K}_{\mathrm{cn}})$ and $c_m^{\mathrm{new}} \longleftarrow h(\mathcal{K}_{\mathrm{cn}} \| \mathrm{id}_w^{\mathrm{new}})$.

Step 3: $\mathcal{SA}$ selects $r_{\mathrm{sa}}^{\mathrm{new}}$ and further computes $\mathrm{tid}_w^{\mathrm{new}} \longleftarrow \mathcal{E}_{K_{\mathrm{cn}}}(\mathrm{id}_w^{\mathrm{new}} \| r_{sa}^{\mathrm{new}})$, and securely stores all the information i.e.,$\{\mathrm{tid}_w^{\mathrm{new}}, \mathcal{L}_{\mathrm{sk}}^{\mathrm{new}}, a_m^{\mathrm{new}}, c_m^{\mathrm{new}}\}$ in $\mathcal{WN}'s$ memory.

Step 4: Finally, it securely stores $\mathcal{L}_{\mathrm{sk}}^{\mathrm{new}}$ to $\mathcal{MN}'s$ memory.

IV. Security Analysis

This section evaluates the proposed protocol's security using formal and informal security analysis techniques.

A. Formal Security Analysis Through $\mathcal{ROR}$ Model

This subsection thoroughly describes the formal security analysis of our protocol by applying the most widely used mathematical $\mathcal{ROR}$ model [18]. The entities in the proposed protocol communication networks are

1) wearable node ($\mathcal{WN}$);
2) mid node ($\mathcal{MN}$);
3) controller node ($\mathcal{CN}$).

The elements of the $\mathcal{ROR}$ model are described below.

$\mathcal{ROR}$ model is based on the $\mathcal{DY}$ and $\mathcal{CK}$ threat models, in which $\mathcal{A}_\mathcal{K}$ has complete authority of the communication channel, for example, $\mathcal{A}_\mathcal{K}$ can snoop, edit, insert, and delete the communications in a network where authenticated devices

communicate with each other. Furthermore, we explain these queries as follows.

1) $\mathcal{EAVES}_{\mathcal{C}}$ $(\Pi^{\mathcal{T}_1}, \Pi^{\mathcal{T}_2}, \Pi^{\mathcal{T}_3})$: An $\mathcal{A}_{\mathcal{K}}$ executes the query to capture all the exchange messages transferred between two authentic participants. This query execution is known as an eavesdropping attack.
2) $\mathcal{REVL}_{\mathcal{C}}$ $(\Pi^{\mathcal{T}})$: An $\mathcal{A}_{\mathcal{K}}$ executes the query to expose $\mathcal{S}_{\mathcal{K}}$ produced by the instance $\Pi^{\mathcal{T}}$ and its partner in the ongoing session.
3) $\mathcal{SND}_{\mathcal{D}}$ $(\Pi^{\mathcal{T}}, \text{msg})$: This query allows $\mathcal{A}_{\mathcal{K}}$ to execute active attacks, where $\mathcal{A}_{\mathcal{K}}$ forwards the message (msg) to $\Pi^{\mathcal{T}}$ and receives a reply message from $\Pi^{\mathcal{T}}$.
4) $\mathcal{CAP}_{\mathcal{MN}}$ $(\Pi^{\mathcal{T}}_{\mathcal{MN}})$: This query allows $\mathcal{A}_{\mathcal{K}}$ to intercept $\mathcal{MN}$ and extracts all secret parameters from $\mathcal{MN}'s$ memory.
5) $\mathcal{CAP}_{\mathcal{WN}}(\Pi^{\mathcal{T}}_{\mathcal{WN}})$: This query allows $\mathcal{A}_{\mathcal{K}}$ to intercept $\mathcal{WN}$ and extracts all secret parameters from $\mathcal{WN}'s$ memory.
6) $\mathcal{TSE}_{\mathcal{T}}(\Pi^{\mathcal{T}})$: This query is designed to explore the semantic securities of $\mathcal{S}_{\mathcal{K}}$ agreed between $\mathcal{WN}$ and $\mathcal{CN}$ following the $\mathcal{ROR}$ model. Here, an unbiased c_n coin is tossed, and its outcome is just known to $\mathcal{A}_{\mathcal{K}}$. If $c_n = 0$, then the query returns a random integer; otherwise, the query returns $(\perp)$ null value.

Theorem: In $\mathcal{ROR}$ model, assume $\mathcal{A}_{\mathcal{K}}$ is an attacker running in the polynomial-time (poly_t) against $\mathcal{P}$, then the benefit of $\mathcal{A}_{\mathcal{K}}$ in successfully extracting the $\mathcal{S}_{\mathcal{K}}$ between $\mathcal{WN}$ and $\mathcal{CN}$ to break the secure semantic strength is denoted as $\mathcal{ADV}^{\mathcal{P}}_{\mathcal{A}_{\mathcal{K}}}(\text{poly}_t) \leq \frac{\mathcal{Q}^2_{\text{hash}}}{|\mathcal{H}|} + \frac{\mathcal{Q}_{\text{send}}}{|\mathcal{P}_{\mathcal{D}}|}$, where $\mathcal{Q}_{\text{send}}$, $\mathcal{Q}_{\text{hash}}$, $|\mathcal{P}_{\mathcal{D}}|$, $|\mathcal{H}|$, and $\mathcal{ADV}^{\mathcal{P}}_{\mathcal{A}_{\mathcal{K}}}(\text{poly}_t)$ denote the number of the sent queries, number of the hash queries, the size of password dictionary, the range space's cryptographic hash method h(.) and $\mathcal{A}_{\mathcal{K}}{}'s$ advantage to break the $\mathcal{P}$ semantic strength in $poly_t$, respectively.

Proof: The theorem's proof presented here is identical to the proof presented in [19]. Moreover, we describe the four games GA_m in our proof, where ($m = 0, 1, 2, 3$). Let $\mathcal{SUC}_i$ be an event in which $\mathcal{A}_{\mathcal{K}}$ correctly guesses the bit c_n in a GA_m; the benefit of successfully winning GA_m by $\mathcal{A}_{\mathcal{K}}$ is expressed as $\mathcal{ADV}^{\text{GA}_m}_{\mathcal{A}_{\mathcal{K}}} = P_r[\mathcal{SUC}_i]$. The following is a comprehensive description of each GA_m.

- $\mathbf{GA}_0$: GA_0 is the real attack in a $\mathcal{ROR}$ model executed by $\mathcal{A}_{\mathcal{K}}$ against $\mathcal{P}$ in which $\mathcal{A}_{\mathcal{K}}$ picks a bit c_n before starting GA_0. According to the definition, the outcome is obtained as

$$\mathcal{ADV}^{\mathcal{P}}_{\mathcal{A}_{\mathcal{K}}}(\text{poly}_t) = |2.P_r[\mathcal{SUC}_0] - 1|. \tag{1}$$

- $\mathbf{GA}_1$: GA_1 is a passive attack executes through $\mathcal{A}_{\mathcal{K}}$ by performing the $\mathcal{EAVES}_C$ $(\Pi^{\mathcal{T}_1}, \Pi^{\mathcal{T}_2}, \Pi^{\mathcal{T}_3})$ query to capture all the transferred messages $\{\text{tid}_w, y_m, a_m, W_n, \mathcal{T}_1\}$, $\{\text{tid}_w, y_m, a_m, V_m, \text{id}_{\text{mn}}, W_n, \mathcal{T}_1, \mathcal{T}_2\}$, $\{\alpha_c, \beta_c, \eta_c, V_c, \mathcal{T}_3\}$, and $\{\alpha_c, \beta_c, \eta_c, V^2_m, \mathcal{T}_4\}$ during the login and authentication phase of our protocol. After intercepting the messages, $\mathcal{A}_{\mathcal{K}}$ executes the $\mathcal{TSE}_{\mathcal{T}}(\Pi^{\mathcal{T}})$ query. The outcome of $\mathcal{TSE}_{\mathcal{T}}(\Pi^{\mathcal{T}})$ is analyzed whether $\mathcal{S}_{\mathcal{K}}$ between $\mathcal{WN}$ and $\mathcal{CN}$ is a random string or a real key. In the proposed protocol, $\mathcal{S}_{\mathcal{K}}$ is calculated as $\mathcal{S}_{\mathcal{K}} \longleftarrow h(\text{id}_w \parallel r_m \parallel f_m \parallel c_m)$, and the captured messages do not expose any secret parameters, i.e., f_m, r_m, c_m, and id_w. So, $\mathcal{A}_{\mathcal{K}}$'s chances of winning the game GA_1 by passive attacks is not expanded. So, it gives

$$P_r[\mathcal{SUC}_1] = P_r[\mathcal{SUC}_0]. \tag{2}$$

- $\mathbf{GA}_2$: GA_2 is an active attack. Using GA_2, $\mathcal{A}_{\mathcal{K}}$ can produce any number of hash queries ($\mathcal{Q}_{\text{hash}}$) for causing hash collisions. Moreover, all the communication messages consist of fresh timestamps and a random number. It is infeasible to generate hash collision occurrence in polynomial time by executing $\mathcal{Q}_{\text{hash}}$ and sending query ($\mathcal{Q}_{\text{send}}$). Thus, applying the birthday contradiction, the subsequent outcome is obtained as follows:

$$|P_r[\mathcal{SUC}_2] - P_r[\mathcal{SUC}_1]| \leq \frac{\mathcal{Q}^2_{\text{hash}}}{2.|\mathcal{H}|}. \tag{3}$$

- $\mathbf{GA}_3$: Through this game, $\mathcal{A}_{\mathcal{K}}$ executes a node capture attack by performing $\mathcal{CAPN}(\Pi^{\mathcal{T}})$ query and extracting all of the secret credentials contained in it. However, it is classified into two components.
  1) When $\mathcal{A}_{\mathcal{K}}$ seizes $\mathcal{MN}$ and utilizes all the secret information. $\mathcal{MN}$ does not retain any secret parameter of $\mathcal{WN}$ or $\mathcal{CN}$. Therefore, no information is attained from GA_2.
  2) When $\mathcal{A}_{\mathcal{K}}$ captures $\mathcal{WN}$ and utilizes all of its information. $\mathcal{CN}$'s secret key $\mathcal{K}_{\text{cn}}$ is not kept in $\mathcal{WN}$.

Hence, we get the result

$$|P_r[\mathcal{SUC}_3] - P_r[\mathcal{SUC}_2]| \leq \frac{\mathcal{Q}_{\text{send}}}{2.|\mathcal{P}_{\mathcal{D}}|}. \tag{4}$$

All *random oracle* queries are performed by $\mathcal{A}_{\mathcal{K}}$ to breach the secure semantic strength of $\mathcal{P}$, $\mathcal{A}_{\mathcal{K}}$ can just guess the c_n bit at the end. It provides $|P_r[\mathcal{SUC}_3]| = \frac{1}{2}$. ■

By applying (1) and (2), we have the following outcome for the GA_m game:

$$\begin{aligned}\mathcal{ADV}^{\mathcal{P}}_{\mathcal{A}_{\mathcal{K}}}(\text{poly}_t) &= |2.P_r[\mathcal{SUC}_0] - 1| \\ &= |2.P_r[\mathcal{SUC}_1] - 1| = 2.|P_r[\mathcal{SUC}_1] - \frac{1}{2}| \\ &= 2.|P_r[\mathcal{SUC}_1] - P_r[\mathcal{SUC}_3]| \end{aligned} \tag{5}$$

By using triangular inequality's concept, we have

$$\begin{aligned}|P_r[\mathcal{SUC}_1] - P_r[\mathcal{SUC}_3]| \leq |P_r[\mathcal{SUC}_1] - P_r[\mathcal{SUC}_2]| \\ + |P_r[\mathcal{SUC}_2] - P_r[\mathcal{SUC}_3]|\end{aligned}$$

Using (3)–(5), we have

$$\frac{1}{2}\mathcal{ADV}^{\mathcal{P}}_{\mathcal{A}_{\mathcal{K}}}(\text{poly}_t) \leq \frac{\mathcal{Q}^2_{\text{hash}}}{2.|\mathcal{H}|} + \frac{\mathcal{Q}_{\text{send}}}{2.|\mathcal{P}_{\mathcal{D}}|}$$

Now, multiply both sides by 2, we get our desired result

$$\mathcal{ADV}^{\mathcal{P}}_{\mathcal{A}_{\mathcal{K}}}(\text{poly}_t) \leq \frac{\mathcal{Q}^2_{\text{hash}}}{|\mathcal{H}|} + \frac{\mathcal{Q}_{\text{send}}}{|\mathcal{P}_{\mathcal{D}}|}$$

B. Formal Security Analysis Through AVISPA Tool

In this subsection, we employ the AVISPA tool to formally verify the proposed protocol's security strength and robustness in terms of specific objectives such as protection against replay attack, mutual authentication, etc. It has four

SUMMARY	SUMMARY
SAFE	SAFE
DETAILS	DETAILS
BOUNDED NUMBER OF SESSIONS	BOUNDED NUMBER OF SESSIONS
PROTOCOL	PROTOCOL
/HOME/SALMAN/DOWNLOADS/	/HOME/SALMAN/DOWNLOADS/
SPAN/TESTSUITE/RESULT/	SPAN/TESTSUITE/RESULT/
PROPOSEDSCHEME.if	PROPOSEDSCHEME.if
GOAL	GOAL
as specified	as specified
BACKEND	BACKEND
OFMC	CL-AtSc
COMMENTS	COMMENTS
STATISTICS	STATISTICS
parse-time: 0.00s	Analyzed : 3 states
search-time: 0.11s	Reachable: 0 states
visited	Translation: 0.11 seconds
Nodes: 32 nodes	Computation: 0.00 seconds
depth: 10 plies	

Fig. 4. Results obtained through AVISPA tool.

backends and abstraction-based techniques integrated using a high-level protocol-specific language ($\mathcal{HLPSL}$). The proposed protocols' security verification results utilize on-the-fly model-checker ($\mathcal{OFMC}$) and the constraint logic-based attack searcher (CLAtSe) backend as depicted in Fig. 4, which demonstrates that our protocol is secure. The results of simulation show that our protocol is well-matched with the desired goals. Under the OFMC, the devised protocol run for 624 s to visit 32 nodes with seven piles depth rate. The facts of Fig. 4 show that both backends generated SAFE output that means the proposed protocol can achieve all of the goals.

C. Informal Security Analysis

This section presents a detailed informal analysis that is explained in the subsequent subsections.

1) Prevent Wearable Node Anonymity and Untraceability (ISA1): An attacker $\mathcal{A}_{\mathcal{K}}$ cannot be able to identify the actual identity id_w of a $\mathcal{WN}$. It also cannot detect any $\mathcal{WN}$ by snooping prior messages in the communication network. In the proposed protocol, $\mathcal{WN}$'s request message $\{\mathrm{tid}_w, y_m, a_m, W_n, \mathcal{T}_1\}$ does not expose id_w. Therefore, $\mathcal{CN}$ does not directly transmit updated $\mathrm{tid}_w^{\mathrm{new}}$ to $\mathcal{WN}$ via $\mathcal{MN}$. Moreover, no $\mathcal{A}_{\mathcal{K}}$ can judge to determine that the same $\mathcal{WN}$ initiating two different sessions. Hence, the proposed protocol ensures the anonymity and untraceability of the wearable node.

2) Prevent Wearable Node Impersonation Attack (ISA2): Assume $\mathcal{A}_{\mathcal{K}}$ intercepts the wearable node's request message, i.e., $\{\mathrm{tid}_w, y_m, a_m, W_n, \mathcal{T}_1\}$ and attempts to produce a valid login request message. $\mathcal{A}_{\mathcal{K}}$ requires id_w of a $\mathcal{WN}$ to calculate $x_m \longleftarrow a_m \oplus \mathrm{id}_w$ and $y_m \longleftarrow x_m \oplus r_m$. But unfortunately, $\mathcal{A}_{\mathcal{K}}$ cannot obtain the real identity id_w of a $\mathcal{WN}$ because id_w is not transmitted in the form of plain text across the network. Hence, our protocol resists wearable node impersonation attack.

3) Prevent Wearable Node Physically Capture Attack (ISA3): Assume $\mathcal{A}_{\mathcal{K}}$ is capable of physically capturing one of the wearable nodes and retrieving all the information contained in it, i.e., $\{\mathcal{L}_{sk}, a_m, c_m, \mathrm{tid}_w\}$ by using a power analysis attack. $\mathcal{A}_{\mathcal{K}}$ is still unable to obtain the $\mathcal{CN}$'s secret key $\mathcal{K}_{\mathrm{cn}}$ to decrypt tid_w as $\mathrm{tid}_w \longleftarrow \mathcal{E}_{K_{\mathrm{cn}}}(\mathrm{id}_w \parallel r_{sa})$. As a result, $\mathcal{A}_{\mathcal{K}}$ is unable to determine the real identity id_w of $\mathcal{WN}$ to calculate $W_n \longleftarrow h(\mathrm{id}_w \parallel \mathrm{tid}_w \parallel c_m \parallel r_m \parallel \mathcal{T}_1)$. Therefore, capturing any $\mathcal{WN}$ will not affect any other $\mathcal{WN}$. Hence, our protocol is safe from wearable node physical capturing attack.

4) Prevent Replay Attack (ISA4): In the designed protocol, all entities check the timestamp freshness to expedite the login phase. In case they are valid, then they further proceed; otherwise, they abort the session. Consequently, $\mathcal{A}_k$ is unable to replay all previously transferred messages. Hence, it is proved that our protocol is protected from replay attack.

5) Prevent Mid Node Impersonation Attack (ISA5): Assume $\mathcal{A}_k$ attempts to make a forged packet, i.e., $\{\alpha_c, \beta_c, \eta_c, V_m^2, \mathcal{T}_4\}$ at the APS 9 in authentication phase and forwards the parameter to the $\mathcal{WN}$. First, $\mathcal{WN}$ computes the parameter $V_m^{2^*} \longleftarrow h(\alpha_c \parallel \beta_c \parallel \eta_c \parallel \mathcal{L}_{\mathrm{sk}} \parallel \mathcal{T}_4)$ and checks with V_m^2. However, $\mathcal{A}_k$ does not know the shared secret key $\mathcal{L}_{\mathrm{sk}}$ between $\mathcal{WN}$ and $\mathcal{MN}$. Hence, it will not be possible for $\mathcal{A}_k$ to make a valid challenge message, and $\mathcal{WN}$ drops the forged packet. Therefore, our protocol protects against mid node impersonation attack.

6) Prevent Controller Node Impersonation Attack (ISA6): Suppose $\mathcal{A}_k$ attempts to impersonate as a legal $\mathcal{CN}$ by transmitting a response message $\{\alpha_c, \beta_c, \eta_c\}$ to $\mathcal{WN}$ through $\mathcal{MN}$. To calculate α_c, $\mathcal{A}_k$ needs to know f_m and $\mathcal{K}_{\mathrm{cn}}$ of the $\mathcal{CN}$ to compute as $\alpha_c \longleftarrow h(\mathcal{K}_{\mathrm{cn}} \parallel \mathrm{id}_w) \oplus f_m$. Even after intercepting $\mathcal{MN}$, $\mathcal{A}_k$ does not know $\mathcal{WN}$ secret identity id_w, and, therefore, it is impossible to execute this attack. As a result, the proposed protocol is resilient against controller node impersonation attack.

7) Prevent Forward/Backward Secrecy (ISA7): In our protocol, $\mathcal{A}_k$ will be unable to calculate id_w, r_m, f_m, c_m, or x_m from $\mathcal{S}_{\mathcal{K}}$ due to securities of the cryptography hash method. Moreover, r_m and f_m are selected randomly, and tid_w and $\mathrm{tid}_w^{\mathrm{new}}$ are updated in each authentication phase. Hence, our protocol meets the features of perfect forward/backward secrecy.

TABLE III
SIMULATION PARAMETERS

Parameters	Explanations
Platform	Ubuntu 16.04 LTS
Standard	$\mathcal{IEEE}$ 802.15.4
Scenarios taken	1, 2, 3
Total $\mathcal{MN}$	1 for all scenarios
Range of $\mathcal{MN}$	100m
Total $\mathcal{WN}$	4, 8, 12 for different scenarios like (1, 2, 3)
Range of $\mathcal{WN}$	20 m
Total $\mathcal{CN}$	1 for all scenarios
Range of $\mathcal{CN}$	50 m
Execution time	1600 s

V. PRACTICAL DEMONSTRATION THROUGH SIMULATOR NS-3

This section presents a detailed discussion of the designed protocol's practical demonstration from the network perspective using the NS-3 (version ns3.33) simulator.

A. Simulation Environment

Simulation experiments are accomplished on Ubuntu 20.04 (LTS) with the widely used simulator NS-3 (ns-3.33 version). The simulation parameters used in this practical study are displayed in Table III. The communication ranges of the Mid Node ($\mathcal{MN}$), Controller Node ($\mathcal{CN}$), and Wearable Node ($\mathcal{WN}$) are assumed to be 100, 50, and 20 m, respectively, during the

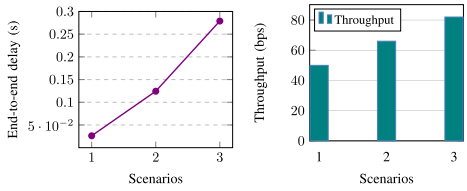

Fig. 5. Simulation results.

simulation. On the other hand, the designed protocol is run for 1600 s.

We have assumed three scenarios for the practical study. Four wearable devices are linked to a single $\mathcal{MN}$ and $\mathcal{CN}$ in the first scenario. Furthermore, the second scenario includes eight $\mathcal{WN}$s connected to a $\mathcal{MN}$ and $\mathcal{CN}$. However, we used 12 $\mathcal{WN}$, 1 $\mathcal{MN}$, and $\mathcal{CN}$ in the third scenario. It is critical to mention that various $\mathcal{WN}$ operate as wearable node, whereas $\mathcal{MN}$ act as a sink and $\mathcal{CN}$ behaves as a sync node.

B. Discussion on Simulation Results

We calculated and evaluated two critical network parameters for the designed protocol, which are detailed below.

1) End-to-End Delay: The end-to-end $\mathcal{ETE}$ delay can be defined as the average transmission time required to reach from the source to the destination. The formula $\mathcal{ETE} = \sum_{i}^{\text{pkt}} \frac{T_{\text{rcv}} \times T_{\text{snd}}}{n}$, where T_{rcv} indicates the received time of the packet. T_{snd} represents the transmitted time, and n represents the total number of packets. The simulation results on the designed protocol under scenarios 1, 2, and 3 are 0.0262, 0.1243, and 0.2789 s, respectively. End-to-end delay facts are also depicted in Fig. 5. The end-to-end delay of the designed protocol grows as the number of $\mathcal{WN}$ increases.

2) Throughput: Throughput is the estimated number of bits delivered per unit time through a communication network. The throughput is denoted as $\mathcal{T} = \sum \frac{\mathcal{N}_m \times \text{Len}_m}{\mathcal{T}_m}$, where $\mathcal{N}_m$ is the total number of packets, Len_m represents the size of the packets, and $\mathcal{T}_m$ indicates the total execution time. The proposed protocol is accomplished under three different scenarios for 1600 s. The throughput results for scenarios 1, 2, and 3 are 50, 66, and 82 bps, respectively. These facts are illustrated in Fig. 5. It is evident through Fig. 5 that the throughput values increase with respect to the scenarios.

VI. Performance Evaluation

This section shows the performance evaluation of the proposed protocol with other related competing protocols [11], [12], [20], [21].

A. Experimental Setup

The proposed protocol and other competing protocols are implemented with three participants, such as $\mathcal{WN}$, $\mathcal{MN}$, and $\mathcal{CN}$. To compute the computation cost, all essential cryptographic operations in $\mathcal{WN}$ are executed on the Arduino device. Moreover, the cryptographic operations of $\mathcal{MN}$ and $\mathcal{CN}$ are executed on mobile device and desktop system, respectively. The specifications for a desktop system, mobile device, and Arduino device are displayed in Table IV. Furthermore, Table V shows the execution time for each cryptographic operation in its implementation context.

TABLE IV
Specifications

Items	Arduino device	Mobile device	Desktop device
Platform	_	Android 9.0 (Pie)	Windows
Clock speed	16 (MHZ)	(4 × 1.8 (GHZ) 260 Silver Kryo + 4 × 2.0 (GHZ) 260 Gold Kryo)	2.9 (GHZ)
System	Microcontroller:(ATmega 328)	Octa Core	Intel Core i7
RAM	2 kB (ATmega 328P)	8 GB	16 GB
IDE	Arduino IDE	_	PyCharm

TABLE V
Execution Time of Cryptographic Operations

Operations	Execution time		
	Arduino	Mobile device	Desktop system
$\mathcal{T}_{\text{enc/dec}}$	0.848 ms	0.02415 ms	0.0023 ms
$\mathcal{T}_{\text{owh}}$	1.184 ms	0.01163 ms	0.0013 ms

TABLE VI
Computation Comparison

Protocols	$\mathcal{WN}$	$\mathcal{MN}$	$\mathcal{CN}$	Total cost
Our	$5\mathcal{T}_{\text{owh}} \approx 5.92$ ms	$3\mathcal{T}_{\text{owh}} \approx 0.03489$ ms	$10\mathcal{T}_{\text{owh}} + 2\mathcal{T}_{enc/dec} \approx 0.0176$ ms	5.97249 ms
Narwal [20]	$6\mathcal{T}_{\text{owh}} \approx 7.104$ ms	$4\mathcal{T}_{\text{owh}} \approx 0.04652$ ms	$11\mathcal{T}_{\text{owh}} \approx 0.0143$ ms	7.16482 ms
Gupta [11]	$7\mathcal{T}_{\text{owh}} \approx 8.288$ ms	$4\mathcal{T}_{\text{owh}} \approx 0.04652$ ms	$13\mathcal{T}_{\text{owh}} \approx 0.0169$ ms	8.35142 ms
Hussain [12]	$6\mathcal{T}_{\text{owh}} \approx 7.104$ ms	$9\mathcal{T}_{\text{owh}} \approx 0.10467$ ms	$5\mathcal{T}_{\text{owh}} \approx 0.0065$ ms	7.21517 ms
Fotouhi [21]	$6\mathcal{T}_{\text{owh}} \approx 7.104$ ms	$21\mathcal{T}_{\text{owh}} \approx 0.24423$ ms	$10\mathcal{T}_{\text{owh}} \approx 0.013$ ms	7.36123 ms

B. Computation Overhead

To determine the computation overhead, we have considered only the cryptographic operations such as symmetric encryption/decryption $\mathcal{T}_{\text{enc/dec}}$, and hash function $\mathcal{T}_{\text{owh}}$. The computation overhead is computed at each side, i.e., $\mathcal{WN}$, $\mathcal{MN}$, and $\mathcal{CN}$. In the proposed protocol, the hash function is executed five times by $\mathcal{WN}$. So, the computation overhead for $\mathcal{WN}$ is 5.92 ms. Similarly, the encryption/decryption and hash functions are executed two times and ten times for $\mathcal{CN}$, while the hash function is executed three times at $\mathcal{MN}$. So, the computation overhead for $\mathcal{CN}$ and $\mathcal{MN}$ are 0.0176 and 0.03489 ms, respectively. Thus, the total computation overhead of the proposed protocol is 5.97249 ms. The computation overhead of all competing protocols [11], [12], [20], [21] is also computed using the same method and displayed in Table VI.

C. Communication Overhead

We examine the communication messages of the login and authentication phase to calculate the communication cost. To compute it, we consider the size of the random number, XOR operation, timestamp, point multiplication, and password to be 160 b each. Although symmetric encryption/decryption consumes 128 b, $\mathcal{CN}$ secret/public keys and hash function consumes 256 b. In the proposed protocol, the participants $\mathcal{WN}$, $\mathcal{MN}$, and $\mathcal{CN}$ exchange four communication messages (i.e., $\mathbb{M}_1$, $\mathbb{M}_2$, $\mathbb{M}_3$, and $\mathbb{M}_4$) among each other. Thus, the number of bits necessary to exchange these messages are 160 + 256 + 256 + 256 + 160 = 1088, 160 + 256 + 256 + 256 + 160 + 256 + 160 + 160 = 1664, 256 + 256 + 256 + 256 + 160 = 1184, and 256 + 256 + 256 + 256 + 160 = 1184, respectively. Therefore, the total communication overhead of our protocol is (1088 + 1664 + 1184 + 1184) = 5120

TABLE VII
COMMUNICATION COMPARISON

Protocols	$\mathcal{WN}$	$\mathcal{MN}$	$\mathcal{CN}$	Total cost
Our	1088 b	2848 b	1184 b	5120 b
[20]	1440 b	3040 b	1440 b	5920 b
[11]	1440 b	3040 b	1440 b	5920 b
[12]	736 b	1440 b	2560 b	4736 b
[21]	768 b	2624 b	1088 b	4480 b

TABLE VIII
SECURITY FEATURES COMPARISON

Protocols	ISA1	ISA2	ISA3	ISA4	ISA5	ISA6	ISA7	ISA8
Our	✓	✓	✓	✓	✓	✓	✓	✓
[20]	✓	✓	✓	✓	✓	✗	✓	✓
[11]	✗	✓	✗	✓	✓	✗	✓	✓
[12]	✓	✗	N/A	N/A	✗	✗	✗	N/A
[21]	✓	✗	✗	N/A	✗	✗	✗	✓

✓: provides; ✗: does not provide; N/A: not applicable.

b. The communication overhead of all competing protocols [11], [12], [20], [21] is also computed using the same method and displayed in Table VII.

D. Security Comparison

The security attributes of the proposed and competing protocols [11], [12], [20], [21] are compared in Table VIII. In Table VIII, our protocol assures all the essential security attributes, while the competing protocols [11], [12], [20], [21] do not prevent wearable node anonymity and untraceability, wearable node physically capture attack, replay attack, impersonation attacks, etc.

VII. CONCLUSION

This article proposed a lightweight authentication protocol for WBAN. We demonstrated that the proposed protocol overcomes security susceptibilities using the formal verification ROM model. In addition, the informal security analysis demonstrated the robustness of the proposed protocol against all related known security threats. We compared the performance of our proposed protocol with other related competing protocols in terms of security attributes, computing, and communication overheads. Hence, the proposed protocol is applicable for WBAN.

In the future, we are hoping to evaluate our protocol by deploying it in a real-world WBAN that will help us fine-tune the designed protocol to measure its performance better. Such efforts will also allow us to develop an energy-efficient protocol that would operate on low battery power.

REFERENCES

[1] J. Li, N. Zhang, J. Ni, J. Chen, and R. Du, "Secure and lightweight authentication with key agreement for smart wearable systems," *IEEE Internet Things J.*, vol. 7, no. 8, pp. 7334–7344, Aug. 2020.

[2] H. U. Rahman, A. Ghani, I. Khan, N. Ahmad, S. Vimal, and M. Bilal, "Improving network efficiency in wireless body area networks using dual forwarder selection technique," *Pers. Ubiquitous Comput.*, vol. 26, no. 1, pp. 11–24, 2022.

[3] T. G. Zimmerman, "Personal area networks: Near-field intrabody communication," *IBM Syst. J.*, vol. 35, no. 3.4, pp. 609–617, 1996.

[4] F. Cherifi, M. Omar, T. Chenache, and S. Radji, "Efficient and lightweight protocol for anti-jamming communications in wireless body area networks," *Comput. Elect. Eng.*, vol. 98, 2022, Art. no. 107698.

[5] S. Shamshad, K. Mahmood, S. Hussain, S. G. A. K. Das, and N. K. J. J. Rodrigues, "An efficient privacy-preserving authenticated key establishment protocol for health monitoring in industrial cyber-physical systems," *IEEE Internet Things J.*, vol. 9, no. 7, pp. 5142–5149, Apr. 2022.

[6] J. Ryu et al., "Secure ECC-based three-factor mutual authentication protocol for telecare medical information system," *IEEE Access*, vol. 10, pp. 11511–11526, 2022.

[7] C.-M. Chen, B. Xiang, T.-Y. Wu, and K.-H. Wang, "An anonymous mutual authenticated key agreement scheme for wearable sensors in wireless body area networks," *Appl. Sci.*, vol. 8, no. 7, 2018, Art. no. 1074.

[8] A. M. Koya and P. Deepthi, "Anonymous hybrid mutual authentication and key agreement scheme for wireless body area network," *Comput. Netw.*, vol. 140, pp. 138–151, 2018.

[9] M. Kompara, S. H. Islam, and M. Hölbl, "A robust and efficient mutual authentication and key agreement scheme with untraceability for WBANs," *Comput. Netw.*, vol. 148, pp. 196–213, 2019.

[10] K. Sowjanya, M. Dasgupta, and S. Ray, "An elliptic curve cryptography based enhanced anonymous authentication protocol for wearable health monitoring systems," *Int. J. Inf. Secur.*, vol. 19, no. 1, pp. 129–146, 2020.

[11] A. Gupta, M. Tripathi, and A. Sharma, "A provably secure and efficient anonymous mutual authentication and key agreement protocol for wearable devices in WBAN," *Comput. Commun.*, vol. 160, pp. 311–325, 2020.

[12] S. J. Hussain, M. Irfan, N. Jhanjhi, K. Hussain, and M. Humayun, "Performance enhancement in wireless body area networks with secure communication," *Wireless Pers. Commun.*, vol. 116, no. 1, pp. 1–22, 2021.

[13] D. Dolev and A. Yao, "On the security of public key protocols," *IEEE Trans. Inf. Theory*, vol. 29, no. 2, pp. 198–208, Mar. 1983.

[14] A. K. Das, M. Wazid, N. Kumar, M. K. Khan, K.-K. R. Choo, and Y. Park, "Design of secure and lightweight authentication protocol for wearable devices environment," *IEEE J. Biomed. Health Inform.*, vol. 22, no. 4, pp. 1310–1322, Jul. 2018.

[15] P. Vijayakumar, M. S. Obaidat, M. Azees, S. H. Islam, and N. Kumar, "Efficient and secure anonymous authentication with location privacy for IoT-based WBANs," *IEEE Trans. Ind. Inform.*, vol. 16, no. 4, pp. 2603–2611, Apr. 2019.

[16] K. Mahmood et al., "Cloud-assisted secure and cost-effective authenticated solution for remote wearable health monitoring system," *IEEE Trans. Netw. Sci. Eng.*, Apr. 2022, doi: 10.1109/TNSE.2022.3164936.

[17] T. Maitra, M. S. Obaidat, R. Amin, S. H. Islam, S. A. Chaudhry, and D. Giri, "A robust ELGamal-based password-authentication protocol using smart card for client-server communication," *Int. J. Commun. Syst.*, vol. 30, no. 11, 2017, Art. no. e3242.

[18] S. Paliwal, "Hash-based conditional privacy preserving authentication and key exchange protocol suitable for industrial Internet of Things," *IEEE Access*, vol. 7, pp. 136073–136093, 2019.

[19] J. Srinivas, A. K. Das, N. Kumar, and J. J. Rodrigues, "Cloud centric authentication for wearable healthcare monitoring system," *IEEE Trans. Dependable Secure Comput.*, vol. 17, no. 5, pp. 942–956, Sep./Oct. 2020.

[20] B. Narwal and A. K. Mohapatra, "SAMAKA: Secure and anonymous mutual authentication and key agreement scheme for wireless body area networks," *Arabian J. Sci. Eng.*, vol. 46, pp. 9197–9219, 2021.

[21] M. Fotouhi, M. Bayat, A. K. Das, H. A. N. Far, S. M. Pournaghi, and M.-A. Doostari, "A lightweight and secure two-factor authentication scheme for wireless body area networks in health-care IoT," *Comput. Netw.*, vol. 177, 2020, Art. no. 107333.