IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received December 31, 2021, accepted January 19, 2022, date of publication January 25, 2022, date of current version February 1, 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3145959

Secure ECC-Based Three-Factor Mutual Authentication Protocol for Telecare Medical Information System

JONGSEOK RYU[1], JIHYEON OH[1], DEOKKYU KWON[1], SEUNGHWAN SON[1], JOONYOUNG LEE[1], YOHAN PARK[2], AND YOUNGHO PARK[1,3], (Member, IEEE)

[1]School of Electronic and Electrical Engineering, Kyungpook National University, Daegu 41566, South Korea
[2]School of Computer Engineering, Keimyung University, Daegu 42601, South Korea
[3]School of Electronics Engineering, Kyungpook National University, Daegu 41566, South Korea

Corresponding authors: Yohan Park (yhpark@kmu.ac.kr) and Youngho Park (parkyh@knu.ac.kr)

This work was supported in part by Kyungpook National University Research Fund, in 2021; and in part by the Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education under Grant 2021R1I1A3059551.

ABSTRACT In the recent COVID-19 situation, Telecare Medical Information System (TMIS) is attracting attention. TMIS is one of the technologies used in Wireless Body Area Network (WBAN) and can provide patients with a variety of remote healthcare services. In TMIS environments, sensitive data of patients are communicated via an open channel. An adversary may attempt various security attacks including impersonation, replay, and forgery attacks. Therefore, numberous authentication schemes have been suggested to provide secure communication for TMIS. Sahoo *et al.* proposed a mutual authentication scheme based on biometrics and Elliptic Curve Cryptography (ECC) in 2020. However, we find out that Sahoo *et al.*'s scheme cannot resist insider and privileged insider attacks and cannot guarantee patient anonymity. In this paper, we propose a secure ECC-based three-factor mutual authentication protocol that guarantees the privacy of patients for TMIS. We conduct informal security analysis to prove that our protocol is secure from various security attacks. In addition, we perform formal security analyses using the Automated Validation of Internet Security Protocols and Applications (AVISPA), Burrows-Abadi-Needham (BAN) logic, and the Real-Or-Random (ROR) model. Furthermore, we assess our protocol's performance and compare it to other protocols. As a result, our protocol has lower communication costs, and better security features compared to related existing protocols. Therefore, our protocol is more appropriate for TMIS environments than other related protocols.

INDEX TERMS Telecare medical information system, authentication, elliptic curve cryptography, biohashing, BAN logic, ROR, AVISPA.

I. INTRODUCTION

In the recent COVID-19 situation, people are increasingly interested in remote services to avoid contact with others. They are hesitant to visit hospitals for fear of contracting COVID-19 from suspected COVID-19 patients. Furthermore, some people may find it difficult to visit the hospital due to their physical condition or personal situation. Therefore, the demand for remote healthcare services including remote diagnosis, prescriptions, and healthcare monitoring is increasing. With the rapid advancement of internet and wireless communication technologies, Wireless Body Area Network (WBAN) is being used efficiently in remote healthcare services. Telecare Medical Information System (TMIS) is one of the technologies used in WBAN and can provide various healthcare services to remote patients via telecare servers [1], [2]. In the COVID-19 situation, TMIS is getting more attention than previous face-to-face healthcare services.

In the TMIS environment, patients can collect their medical information including the heart rate, blood pressure, and body temperature by wearable sensor devices. Then, medical information is transmitted to their mobile devices. Following that, patients can transmit the collected medical information to telecare servers at any time and from any location. Telecare servers provide proper healthcare services including medical monitoring, treatment, and prescription to patients after

The associate editor coordinating the review of this manuscript and approving it for publication was Ilsun You.

receiving their medical information. Therefore, patients can reduce time and cost consumption and utilize the various healthcare services at home. Because of these benefits, TMIS provides more suitable healthcare services than existing face-to-face healthcare in the COVID-19 situation.

Despite the above advantages, TMIS has several problems with the security aspect. In TMIS, the telecare server is maintaining the privacy and medical information of patients including identity, password, and electronic medical records. Only legitimate patients should be able to access their own medical information in order to protect their privacy and the secrecy of their medical information. Furthermore, sensitive patient data is transmitted to the telecare server via an insecure channel so that an adversary can attempt security attacks including impersonation, replay, and man-in-the-middle (MITM) attacks. As a result, secure mutual authentication and key agreement protocols are critical issues in TMIS environments. Recently, many studies have been proposed to handle TMIS security flaws [3], [4].

In 2020, Sahoo *et al.* [4] designed a mutual authentication protocol for TMIS using IoT-enabled devices. To protect sensitive patient data, their protocol utilizes biometric information, symmetric cryptography, and Elliptic Curve Cryptography (ECC). They claimed that their protocol can withstand a variety of security attacks including stolen smartcards, offline password guessing, and replay attacks. However, we found out that their protocol is still vulnerable to insider and privileged insider attacks. We also found that their protocol cannot provide patient anonymity and correct password update phase. In this paper, we propose a secure ECC-based three-factor mutual authentication protocol for TMIS using a mobile device of patients.

A. RESEARCH CONTRIBUTIONS

The contributions of this paper are represented below.

- We review Sahoo *et al.*'s protocol to show that their protocol is vulnerable to insider and privileged insider attacks. We also demonstrate that their protocol cannot provide patient anonymity and correct password change phase.
- We propose a secure mutual authentication protocol using biometrics and ECC to guarantee secure communication for TMIS environments. Then, we conduct an informal security analysis to demonstrate that our protocol is resistant to various security attacks including insider, privileged insider, and stolen mobile device attacks, and can provide patient anonymity.
- We perform formal security analyses using the BAN logic [5] to show mutual authentication and the ROR model [6] to show session key security. In addition, we conduct the AVISPA [7], [8] to demonstrate that our protocol can prevent replay and MITM attacks.
- We evaluate the computation costs, communication costs, and security features of our protocol. As a result, our protocol provides lower communication costs, and better security compared with related existing protocols.

B. ORGANIZATION

The related work is presented in Section II. The preliminaries including the ECC, Biohashing, adversary model, and system model are explained in Section III. We provide a review of Sahoo *et al.*'s protocol and cryptanalysis of their protocol in Sections IV and V. Then, we proposed the secure authentication protocol in Section VI. The security and performance analyses of our protocol are shown in Sections VII and VIII. Finally, we presented the paper's conclusion in Section IX.

II. RELATED WORK

In the past few years, many authentication schemes have been proposed for TMIS environments. In 2013, Khan and Kumari [9] suggested two-factor based authentication using smartcard for TMIS. They said that their scheme is secure against offline password guessing, replay, and stolen verifier attacks. However, their scheme cannot resist the offline password guessing attack. Girl *et al.* [10] proposed a user authentication scheme based on RSA in 2014 to improve Khan and Kumari's scheme. Through informal analysis, Girl *et al.* claimed that their scheme is secure from various security attacks such as insider, replay, and offline password guessing attacks. In 2015, Amin and Biswas [11] discovered that Girl *et al.*'s scheme is vulnerable to offline password guessing and privileged insider attacks. Thereafter, they suggested an improved RSA-based authentication scheme. They also conduct the AVISPA to validate its security. Nevertheless, Sutrala *et al.* citer12 discovered that Amin and Biswas's scheme cannot withstand offline password guessing, impersonation, and replay attacks. Following that, they proposed authentication and key agreement scheme based on RSA.

In 2015, Zhang and Zhu [13] proposed an authenticated key agreement scheme for TMIS using ECC that provides a higher security level with a lower key size than RSA. They asserted that their scheme is resistant to MITM and offline password guessing attacks. However, Liu *et al.* [14] discovered that Zhang and Zhu's scheme is still vulnerable to offline password guessing and stolen smartcard attacks. Ostad-Sharif *et al.* [15] also proposed an ECC-based authentication scheme in 2018. Despite being more efficient than RSA, their scheme has security flaws such as key compromise impersonation and password guessing attacks [16]. These schemes [9]–[15] rely on the password and smartcard as factors, and they cannot protect against offline password guessing or stolen smartcard attacks. As a result, two-factor schemes are inappropriate in TMIS environments.

To improve the security flaws of two-factor in the TMIS environment, many researchers proposed a three-factor authentication scheme [4], [17]–[23]. In 2015, Lu *et al.* [17] suggested an authentication scheme using biometrics for the TMIS environment. They asserted that their scheme can withstand a variety of security attacks including offline password guessing and replay attacks. However, their scheme cannot withstand offline password guessing and impersonation attacks [18]. In 2016, Ravanbakhsh and Nazari [19]

suggested an improvement in mutual authentication and a session key agreement scheme for TMIS. Unfortunately, Ostad-Sharif *et al.* [20] proved that Ravanbakhsh and Nazari's scheme cannot prevent known session-specific temporary information attacks and cannot ensure perfect forward secrecy. Following that, Qi and Chen [21] proposed mutual authentication for TMIS using biometrics and ECC in 2018. By using BAN logic, they demonstrated that their scheme provides mutual authentication. Nonetheless, Qi and Chen scheme suffers from security flaws including offline password guessing and key compromise impersonation attacks [22]. As a result, [17], [19], and [21] are still not ideal for TMIS environments.

In 2020, Sahoo *et al.* [4] designed a three-factor authentication scheme to address security flaws of related existing schemes for the TMIS environment. They claimed that their scheme can withstand security attacks including stolen smartcard, offline password guessing, and insider attacks. However, we discovered that Sahoo et al's scheme is still vulnerable to insider and privileged insider attacks. We also found that their scheme cannot ensure patient anonymity and has a flaw in the password update phase. Therefore, we propose a secure mutual authentication scheme for TMIS security using biometrics and ECC.

III. PRELIMINARIES

In this section, we explain the basic concept including ECC and biohashing. We also present the adversary model and system model of our protocol.

A. ELLIPTIC CURVE CRYPTOGRAPHY

ECC is public-key cryptography based on the elliptic curve over a large finite field [24]. ECC can provide more security and better performance with smaller key sizes compared to modern public-key cryptography. Let p be a large prime number, $r, s \in F_p$, and $4r^3 + 27s^2 \neq 0 \ (mod \ p)$. Then, a nonsingular elliptic curve $E_p(r, s)$ over a finite field F_p is defined equation as below:

$$E_p(r, s) : y^2 = x^3 + rx + s \ (mod \ p)$$

Let Q be a base point on elliptic curve $E_p(r, s)$. Then, the operation of scalar multiplication is defined as $k \cdot Q = Q + \cdots + Q$ (k times), where $k \in F_p$ is a positive integer. The security of ECC is based on the problems as follows.

- Elliptic Curve Discrete Logarithm Problem (ECDLP). Given two points P, Q on elliptic curve $E_p(r, s)$ such that $Q = k \cdot P$, where $k \in F_p$, it is computationally hard to determine k.
- Elliptic Curve Diffie-Hellman Problem (ECDHP). Given three points P, $x \cdot P$, $y \cdot P$ on elliptic curve $E_p(r, s)$, it is computationally hard to determine $x \cdot y \cdot P$.
- Elliptic Curve Decisional Diffie-Hellman Problem (ECDDHP). Given four points $P, x \cdot P, y \cdot P, z \cdot P$ on elliptic curve $E_p(r, s)$, to decide whether $z \cdot P = x \cdot y \cdot P$ or not, where $x, y, z \in Z_p^*$.

B. BIOHASHING

A user's biometric information is the best way to verify a real user in an authentication system. In 2004, Jin *et al.* [25] proposed a biohashing function based on fingerprint data to identify users. It is based on the inner products of tokenized pseudo-random numbers and fingerprint data from the user. In 2006, Lumini *et al.* [26] present an improved biohashing function. According to the research, the biohashing function converts fingerprint biometric information into a bit vector.

- Extract the biometric feature and represent it as a fixed-length feature vector $x \in R^n$ from the fingerprint.
- Generate the random number $r_i \in R^n (i = 1, \cdots, n)$ by pseudo-random number algorithms Blum–Blum–Shub methods.
- Repeat the Gram-Schmidt ortho-normalization about generated random number and create the orthonormal set of vectors $o_i \in R^n (i = 1, \cdots, n)$.
- Compute the inner product between the biometric feature and orthonormal set of vectors. Finally, compute the BioHash code b_i as

$$b_i = \begin{cases} 0, & if \ \langle x | o_i \rangle \leq \upsilon \\ 1, & if \ \langle x | o_i \rangle > \upsilon, \end{cases}$$

  where υ is a present threshold.

C. ADVERSARY MODEL

We adopt the "Dolev-Yao (DY) model" [27], which is generally used for the analysis of the security protocols. Under the DY model, an adversary can intercept, delete, and modify transmitted messages via an insecure channel. The capabilities of an adversary can be defined as below.

- An adversary can perform impersonation, forgery, and MITM attack, etc. [28].
- An adversary can obtain a mobile device of the legitimate patient and can extract all the stored information in the mobile device using power analysis attacks [29]–[31].
- An adversary can be a legitimate patient or a privileged insider at the registration center.

In addition, we consider the "Canetti-Krawczyk (CK) model" [32], which has a more powerful assumption than the DY model. A malicious adversary can compromise secure information such as the private key, master key, and session secret credentials using the CK model.

D. SYSTEM MODEL

The system model consists of the patients, telecare servers, and a registration center, as indicated in Figure 1. The registration center is a trusted authority and initializes system parameters. The patients must register at the registration center once to access telecare servers. Likewise, telecare servers must register at the registration center for participating TMIS environment. Then, the registration center distributes secret values to the patients and telecare servers for authentication. To use medical service of telecare servers, a patient creates

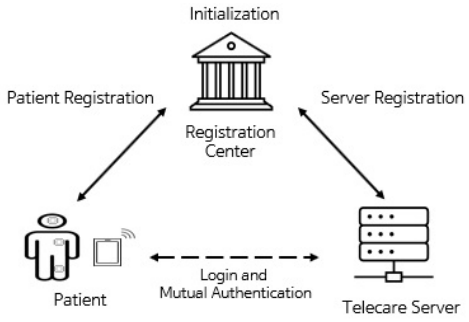

FIGURE 1. The system model of our protocol in TMIS.

login request message using secret values and transmits it to telecare servers. After patient verification, telecare servers compute and send the response message to the patient for authentication. If mutual authentication is successfully finished, the session key is established between patient and telecare servers. Finally, patient and telecare servers can securely exchange sensitive information such as patient's healthcare data during the communication. The detailed descriptions of communication participants are as below.

- Patient: A patient can collect healthcare data such as heart rate, blood pressure, and body temperature by their sensor devices. To utilize the telecare server's medical services, the patient must first register at the registration center using a mobile device. After registration, the patient can use a mobile device to send collected data to telecare servers.
- Telecare server: The telecare server must register at the registration center to be a legitimate entity. After registration, the telecare server receives the patient's collected healthcare data from their mobile devices. Then, the telecare server provides medical services to the patients such as diagnosis and prescriptions. The telecare server has enough storage space to store the patient's information for authentication.
- Registration center: The registration center is a trusted authority that generates system parameters and publishes public information. The registration center is in charge of registration patients and telecare servers. During the registration process, the registration center computes and distributes secret values to patients and telecare servers for authentication.

IV. REVIEW OF SAHOO *et al.*'s PROTOCOL

In this section, we review Sahoo *et al.*'s protocol. They proposed an ECC-based authentication protocol for TMIS, which includes registration phase for patients and telecare servers, login and authentication phase, and password change phase. Table 1 represents the notations of this paper.

A. INITIALIZATION PHASE

In this phase, RC selects an elliptic curve $E_p(r, s)$ and chooses a base point P on $E_p(r, s)$. Afterward, RC selects a master key mk and computes $Pub = mk \cdot P$. Then, RC chooses a pair $\{x, y\}$ and defines $\{x, y, mk\}$ as the private key and $\{E, P, Pub\}$ as the public key.

TABLE 1. Notations of this paper.

Notaion	Definition
U_i	Patient
TS_j	Telecare server
SID_j	Identity of TS_j
$PSID_i$	Pseudo identity of TS_j
RC	Registration center
RCI	Identity of RC
SC	Smartcard
ID_i	Identity of U_i
PID_i	Pseudo identity of U_i
PW_i	Password of U_i
BI_i	Biometric information of U_i
MD_i	Mobile device of U_i
$R_j, RN_u, RN_{sj}, n_1, n_2$	Random numbers
T_u, T_r, T_1, T_2	Timestamps
k_{sj}	Private key of TS_j
pk_{sj}	Public key of TS_j
x, y, k_{rc}	Private key of RC
mk	Master key of RC
SK	Session key
$h(\cdot)$	Hash function
$h_b(\cdot)$	Biohash function
E_k/D_k	Symmetric encryption/decryption
$\oplus$	Exclusive or operation
$\|\|$	Concatenation operation

B. REGISTRATION PHASE

The patient U_i and telecare server TS_j register to RC for participating in the TMIS environment. The detailed protocol of the registration phase is shown in Figure 2.

1) PATIENT REGISTRATION PHASE

U_i registers in RC through a secure channel by the following steps.

1) U_i selects an identity ID_i, a password PW_i, and imprints the biometrics BI_i.
2) U_i computes $Gen(BI_i) = (\sigma_i, \theta_i)$ using fuzzy extractor, and $PW_{i1} = h(PW_i||\sigma_i)$. Then, U_i transmits message $\{ID_i, PW_{i1}, BI_i, SN_i\}$ to RC for registration.
3) RC receives registration request message from U_i. Then, RC calculates $A_i = h(PW_{i1}||BI_i) \cdot P = (P_x, P_y)$, $B_i = h(ID_i||PW_{i1}||BI_i)$, $C_i = SN_i \oplus h(x||y) \oplus h(RCI||x)$, and $G_i = h(RCI||x) \oplus h(ID_i||PW_{i1})$. Afterward, RC stores ID_i, A_i, C_i in its database and issues the smartcard SC with the parameters $\{B_i, C_i, G_i, h(\cdot), E_k, D_k\}$ to U_i.
4) U_i receives SC to RC and stores parameter θ_i in SC. Then, SC includes the parameters $\{B_i, C_i, G_i, h(\cdot), E_k, D_k, \theta_i\}$.

2) TELECARE SERVER REGISTRATION PHASE

TS_j registers from RC through a secure channel. The detailed steps are as below.

1) TS_j chooses a identity SID_j and transmits $\{SID_j\}$ to RC for registration.
2) RC selects a random number R_j and calculates $R_{1j} = h(R_j||h(x||y))$. Finally, RC stores $\{R_{1j}, SID_j\}$ in the database and transmits $\{R_{1j}\}$ to TS_j.

C. LOGIN PHASE

U_i executes the following steps to use medical services.

1) U_i inserts SC into a card reader and inputs ID_i, PW_i, and BI_i. Then, SC computes $\sigma_i^* = Rep(BI_i, \theta_i)$, $PW_{i1}^* = h(PW_i||\sigma_i^*)$, and $B_i^* = h(ID_i||PW_{i1}^*||BI_i)$.
2) SC checks whether $B_i^* \stackrel{?}{=} B_i$. If it is same, move to the next step. Otherwise, SC terminates the login phase.
3) SC selects a random number n_1 and computes $A_i^* = h(PW_{i1}||BI_i) \cdot P = (P_x, P_y)$, $h(RCI||x) = G_i \oplus h(ID_i||PW_{i1}^*)$, $M_1 = n_1 \cdot P$, $M_2 = n_1 \cdot Pub$, $M_3 = n_1 \oplus h(ID_i||C_i||A_i^*||T_u)$, $M_4 = h(ID_i||M_1||M_2||n_1||T_u)$, and $M_5 = E_{(SN_i||h(RCI||x))}(ID_i||M_3||P_y)$. Then, SC sends the login message $\{M_4, M_5, C_i, T_u\}$ to RC via an open channel.

D. AUTHENTICATION PHASE

After receiving the login request message $\{M_4, M_5, C_i, T_u\}$, RC and TS_j perform the following steps.

1) RC checks the validity of T_u by calculating $T_r - T_u \leq \Delta T$. If the timestamp is valid, RC calculates $SN_i^* = h(x||y) \oplus C_i \oplus h(RCI||x)$, $D_{(SN_i||h(RCI||x))}(M_5) = (ID_i||M_3||P_y)$, $n_1^* = M_3 \oplus h(ID_i||C_i||A_i||T_u)$, $M_1^* = n_1^* \cdot P$, $M_2^* = n_1^* \cdot Pub$ and checks $M_4^* \stackrel{?}{=} h(ID_i||M_1^*||M_2^*||n_1^*||T_u)$. If it is equal, RC calculates $M_6 = h(ID_i||SID_j||M_1^*||R_{1j}||T_r)$, and $M_7 = E_{R_{1j}}(ID_i||M_1^*||P_y)$. Then, RC transmits the message $\{M_6, M_7, T_r\}$ to TS_j through a public channel.
2) TS_j verifies the timestamp as $T_s - T_r \leq \Delta T$. If it is vaild, TS_j calculates $D_{R_{1j}}(M_7) = (ID_i||M_1^*||P_y)$ and compares $M_6 = h(ID_i||SID_i||M_1^*||R_{1j}||T_r)$. If it is the same, TS_j selects a random number n_2 and calculates $S_1 = n_2 \cdot P$, $S_2 = h(ID_i||SID_j||P_y)$, $SK = n_2 \cdot M_1^*$, $S_3 = h(ID_i||S_1||SK||T_s)$, $S_4 = E_{h(ID_i||P_y)}(S_1||S_2)$. Afterward, TS_j transmits the message $\{S_3, S_4, T_s\}$ to U_i through a public channel.
3) U_i checks the timestamp by computing $T_u^* - T_r \leq \Delta T$. If the timestamp is vaild, U_i computes $D_{h(ID_i||P_y)}(S_4) = (S_1||S_2)$, $SK^* = n_1 \cdot S_1$, and $S_3^* = h(ID_i||S_1||SK^*||T_s)$. Then, U_i checks whether $S_3^* \stackrel{?}{=} S_3$. If it is equal, U_i computes $M_8 = h(ID_i||S_2||SK^*||T_u^*)$ and sends $\{M_8, T_u^*\}$ to TS_j for authentication.
4) TS_j receives the messages $\{M_8, T_u^*\}$ and checks the timestamp as $T_s^* - T_u^* \leq \Delta T$. If the timestamp is valid, TS_j verifies $M_8 \stackrel{?}{=} h(ID_i||S_2||SK||T_u^*)$. If it is the same, the authentication phase is finished. It indicates that the session key agreement and mutual authentication between U_i and TS_j were successful. Figure 3 depicts the detailed process of the login and authentication phase.

E. PASSWORD CHANGE PHASE

U_i can freely change the old password. The detailed steps are as follows.

1) U_i inserts SC into a card reader, and inputs ID_i, PW_i, BI_i. Afterward, SC calculates $\sigma_i^* = Rep(BI_i, \theta_i)$, $PW_{i1}^* = h(PW_i||\sigma_i^*)$, and $B_i^* = h(ID_i||PW_{i1}^*||BI_i)$. Then, SC checks $B_i^* \stackrel{?}{=} B_i$. If it is same, SC requests U_i's new password PW_i^{new}.
2) After receiving PW_i^{new}, SC calculates $PW_{i1}^{new} = h(PW_i^{new}||\sigma_i)$, $B_i^{new} = h(ID_i||PW_{i1}^{new}||BI_i)$. Finally, SC replaces B_i with B_i^{new}.

V. CRYPTANALYSIS OF SAHOO *et al.*'s PROTOCOL

In this section, we analyze the security flaws of Sahoo *et al.*'s protocol. We demonstrate that their protocol is vulnerable to insider and privileged insider attacks. Furthermore, their protocol cannot provide patient anonymity and has a flaw in the patient's password update.

A. INSIDER ATTACK

Suppose that a malicious adversary U_a registers to the registration center RC as a legitimate patient. Thereafter, U_a can imitate a legal patient U_i. The detailed steps are as below.

1) U_a selects a sensor SN_a and registers in RC with U_a's ID_a, PW_a, and BI_a. Thereafter, U_a receives SC from RC and stores θ_a into SC. Then, SC includes the parameters $\{B_a, C_a, G_a, h(\cdot), E_k, D_k, \theta_a\}$.
2) U_a can compute $h(RCI||x)$ and $h(x||y)$ using U_a's information and parameters in SC.
3) U_a obtains the login request message $\{M_4, M_5, C_i, T_u\}$ by eavesdropping attack. Afterward, U_a can compute $SN_i = C_i \oplus h(x||y) \oplus h(RCI||x)$, and $(ID_i||M_3||P_y) = D_{(SN_i||h(RCI||x))}(M_5)$. Then, U_a can extract the parameters ID_i, M_3, and P_y.
4) U_a selects n_{1a} and T_a. Thereafter, U_a computes $M_{1a} = n_{1a} \cdot P$, $M_{2a} = n_{1a} \cdot Pub$, $M_{3a} = n_{1a} \oplus h(ID_i||C_i||A_i||T_a)$, $M_{4a} = h(ID_i||M_{1a}||M_{2a}||n_{1a}||T_a)$, and $M_{5a} = E_{(SN_i||h(RCI||x))}(ID_i||M_{3a}||P_y)$.
5) After computing the parameters, U_a sends the message $\{M_{4a}, M_{5a}, C_i, T_a\}$ to RC for authentication between RC and TS_j. Then, RC and TS_j perform the authentication phase. Finally, U_a can compute SK as a legal patient U_i to access the telecare server TS_j.

Thus, Sahoo *et al.*'s protocol cannot prevent insider attacks through the above steps.

B. PATIENT ANONYMITY

In Section V-A, we showed that U_a can successfully obtain U_i's ID_i by decrypting M_5, where $(ID_i||M_3||P_y) = D_{(SN_i||h(RCI||x))}(M_5)$. Therefore, Sahoo *et al.*'s protocol cannot guarantee patient anonymity.

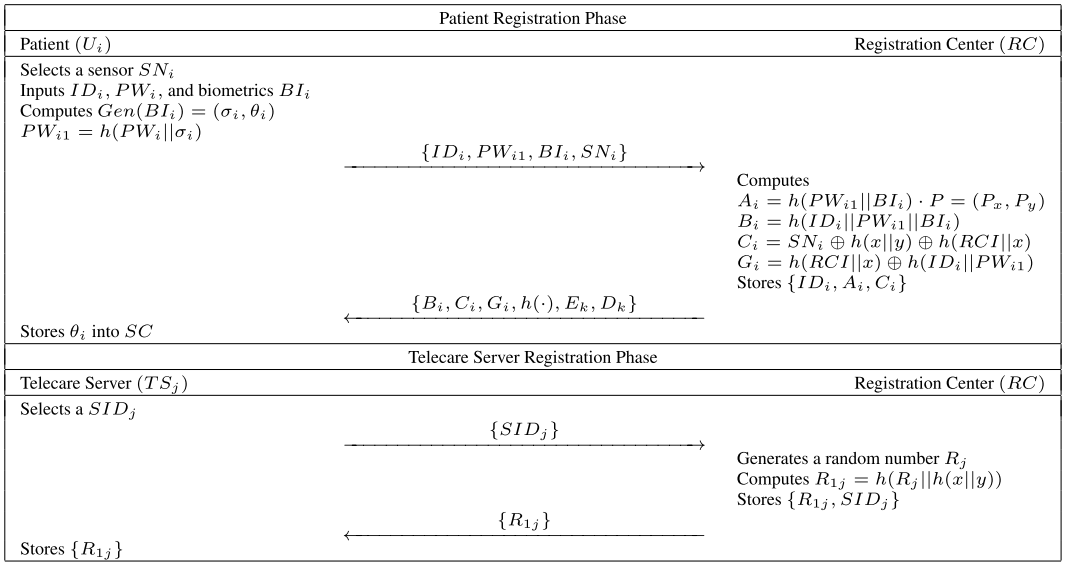

FIGURE 2. Patient and telecare server registration phase of Sahoo *et al.*'s protocol.

C. PRIVILEGED INSIDER ATTACK

Let U_a be a privileged insider user of RC. Then, U_a has U_i's registration information $\{ID_i, PW_{i1}, BI_i, SN_i\}$. Afterward, U_a can successfully calculate SK between U_i and TS_j. The details are described below.

1) According to Section III-C, U_a can obtain SC of U_i and extract the parameters $\{B_i, C_i, G_i, h(\cdot), E_k, D_k\}$ in SC. Also, U_a can obtain the message $\{M_4, M_5, C_i, T_u\}$ by eavesdropping attack.
2) U_a can extract M_3 by decrypting M_5 and compute $A_i^* = h(PW_{i1}||BI_i) \cdot P = (P_x, P_y), n_1 = M_3 \oplus h(ID_i||C_i||A_i^*||T_u), M_1 = n_1 \cdot P, M_2 = n_1 \cdot Pub$. Then, U_a calculates $M_4^* = h(ID_i||M_1||M_2||n_1||T_u)$, $M_5^* = E_{(SN_i||h(ECI||x))}(ID_i||M_3||P_y)$.
3) U_a transmits the message $\{M_4^*, M_5^*, C_i, T_u\}$ to RC. Afterward, RC and TS_j perform the authentication phase and U_a obtains the message $\{S_3, S_4, T_s\}$. Finally, U_a can compute SK between U_i and TS_j.

Therefore, Sahoo *et al.*'s protocol cannot withstand privileged insider attacks.

D. FLAW IN PASSWORD CHANGE PHASE

U_i inserts SC and inputs ID_i, PW_i, and BI_i for changing the password. After verifying the patient credential B_i, SC requests new password PW_i^{new} from U_i. Then, U_i sends PW_i^{new} to SC. Thereafter, SC computes $PW_{i1}^{new} = h(PW_i^{new}||\sigma^*), B_i^{new} = h(ID_i||PW_{i1}^{new}||BI_i)$ and replaces B_i with B_i^{new}. However, G_i and A_i are still made by old value PW_{i1} like as $G_i = h(RCI||x) \oplus h(ID_i||PW_{i1}^*)$ and $A_i = h(PW_{i1}||BI_i) \cdot P$. Therefore, Sahoo *et al.*'s protocol cannot provide correct password change.

VI. PROPOSED PROTOCOL

In this section, we propose a secure ECC-based three-factor mutual authentication protocol for TMIS to address the security flaws in Sahoo *et al.*'s protocol. We also consider the efficiency of authentication phase. The proposed protocol is composed of three phases: patient and telecare server registration, login and authentication, and password change.

A. INITIALIZATION PHASE

In this phase, RC selects an elliptic curve $E_p(r, s)$ over a F_p. Then, RC chooses a base point P on $E_p(r, s)$ and a private key k_{rc}. Afterward, RC publishes the system parameters $\{E_p(r, s), P, h(\cdot), h_b(\cdot)\}$.

B. REGISTRATION PHASE

To participate in the TMIS environment, the patient U_i and telecare server TS_j must register to RC. The details are shown in Figure 4.

1) PATIENT REGISTRATION PHASE

U_i registers in RC to use medical services from TS_j. The details are presented as below.

1) U_i inputs ID_i, PW_i, BI_i and generates a random number RN_u. Afterward, U_i computes $HID_i = h(ID_i||RN_u), HPW_i = h(PW_i||h_b(BI_i))$, and $GPW_i = HPW_i \oplus RN_u$ and sends the message $\{HID_i, GPW_i\}$ to RC via a secure channel.
2) RC computes $UR_i = h(HID_i||k_{rc})$ and $B_i = UR_i \oplus GPW_i$. Afterward, RC stores $\{HID_i\}$ in the database and sends $\{B_i\}$ to U_i.
3) U_i computes $RPW_i = h(ID_i||PW_i||h_b(BI_i)), A_1 = RN_u \oplus RPW_i, A_2 = h(HID_i||HPW_i||RPW_i||RN_u)$, and $A_3 = B_i \oplus RN_u = UR_i \oplus HPW_i$. Then, U_i stores the parameters $\{A_1, A_2, A_3\}$ in the mobile device MD_i

2) TELECARE SERVER REGISTRATION PHASE

TS_j must register in RC to provide medical services for U_i. The detailed steps are given below.

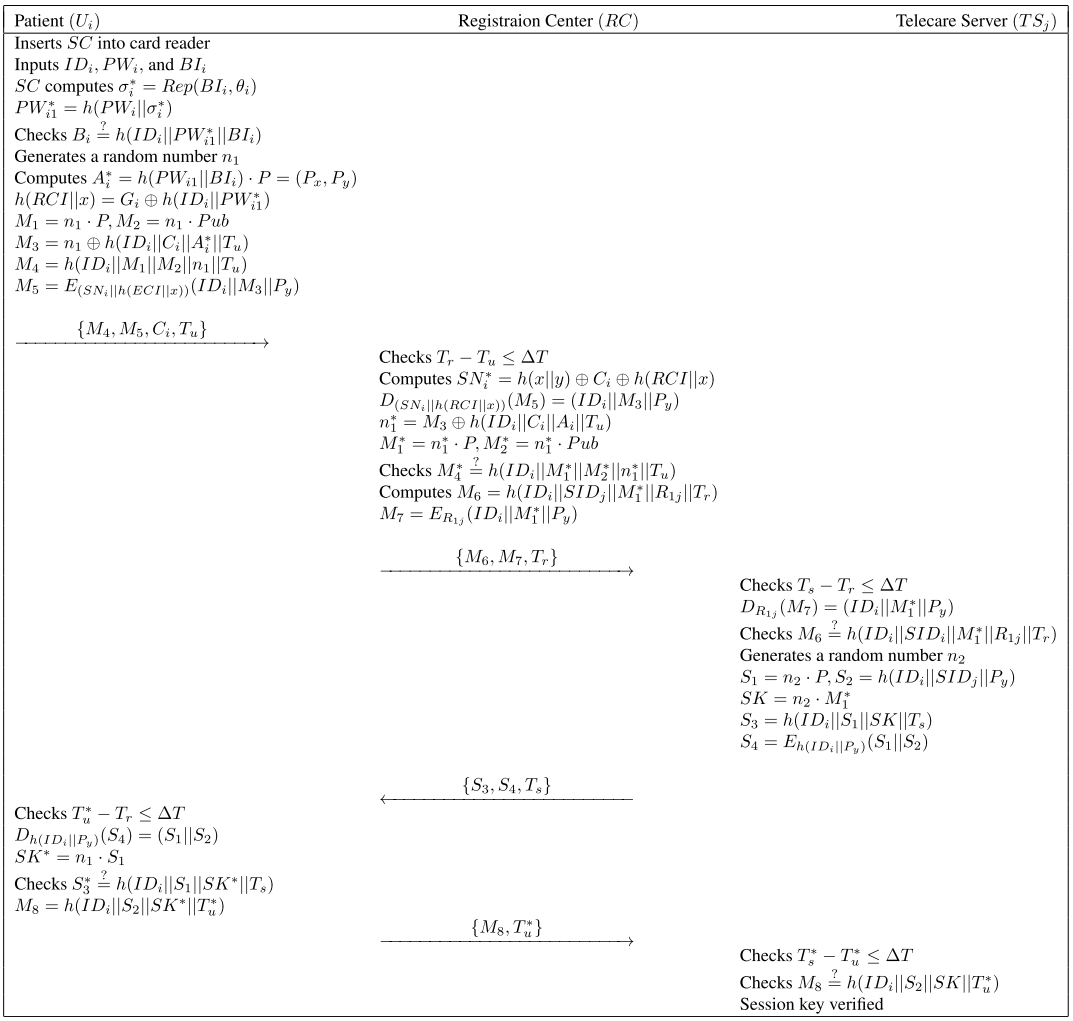

FIGURE 3. Login and authentication phase of Sahoo *et al.*'s protocol.

1) TS_j chooses an identity SID_j and a random number RN_{sj}. Then, TS_j calculates the pseudo identity $PSID_j = SID_j \oplus RN_{sj}$ and transmits $\{PSID_j, RN_{sj}\}$ to RC via a secure channel.
2) RC computes $SID_j = PSID_j \oplus RN_{sj}$ and stores SID_j in the database. Afterward, RC retrieves HID_i in its database and calculates $k_{sj} = h(SID_j||k_{rc}), pk_{sj} = k_{sj} \cdot P, TID_i = h(HID_i||pk_{sj}), UR_i = h(HID_i||k_{rc})$, and $V_{ij} = h(PSID_j||UR_i)$. Thereafter, RC makes $\{PSID_j, pk_{sj}\}$ public and transmits the parameters $\{k_{sj}, TID_i, V_{ij}\}$ to TS_j.
3) TS_j is defined k_{sj} as a private key. Then, TS_j calculates $SV_{ij} = V_{ij} \oplus h(SID_j||k_{sj})$ and stores the parameters $\{TID_i, SV_{ij}\}$ in the database.

C. LOGIN PHASE

U_i executes the following steps to access TS_j for using medical services.

1) U_i inputs ID_i, PW_i, and BI_i in MD_i. Then, MD_i calculates $RPW_i = h(ID_i||PW_i||h_b(BI_i))$, $RN_u = A_1 \oplus RPW_i, HID_i = h(ID_i||RN_u), HPW_i = h(PW_i||h_b(BI_i))$, and $A_2^* = h(HID_i||HPW_i||RPW_i||RN_u)$.
2) MD_i checks whether $A_2^* \stackrel{?}{=} A_2$. If it is equal, move to the next step. Otherwise, MD_i discontinues the login phase.
3) MD_i generates a random number n_1 and a timestamp T_1. Afterward, MD_i computes $S_1 = n_1 \cdot P, S_2 = n_1 \cdot pk_{sj}, UR_i = A_3 \oplus HPW_i, PID_i = h(HID_i||pk_{sj}) \oplus h(PSID_j||S_2), UID_i = h(h(HID_i||pk_{sj})||h(PSID_j||UR_i)||T_1)$, and $M_i = h(UID_i||S_2||h(PSID_j||UR_i)||T_1)$. Then, MD_i sends $\{PID_i, M_i, S_1, T_1\}$ to TS_j via a public channel.

D. AUTHENTICATION PHASE

TS_j performs the following steps to authentication between U_i and TS_j.

1) TS_j checks whether $|T_1^* - T_1| \leq \Delta T$. If it is valid, TS_j calculates $S_2 = k_{sj} \cdot S_1$ and $TID_i = h(HID_i||pk_{sj}) = PID_i \oplus h(PSID_j||S_2)$. Thereafter, TS_j retrieves SV_{ij} in

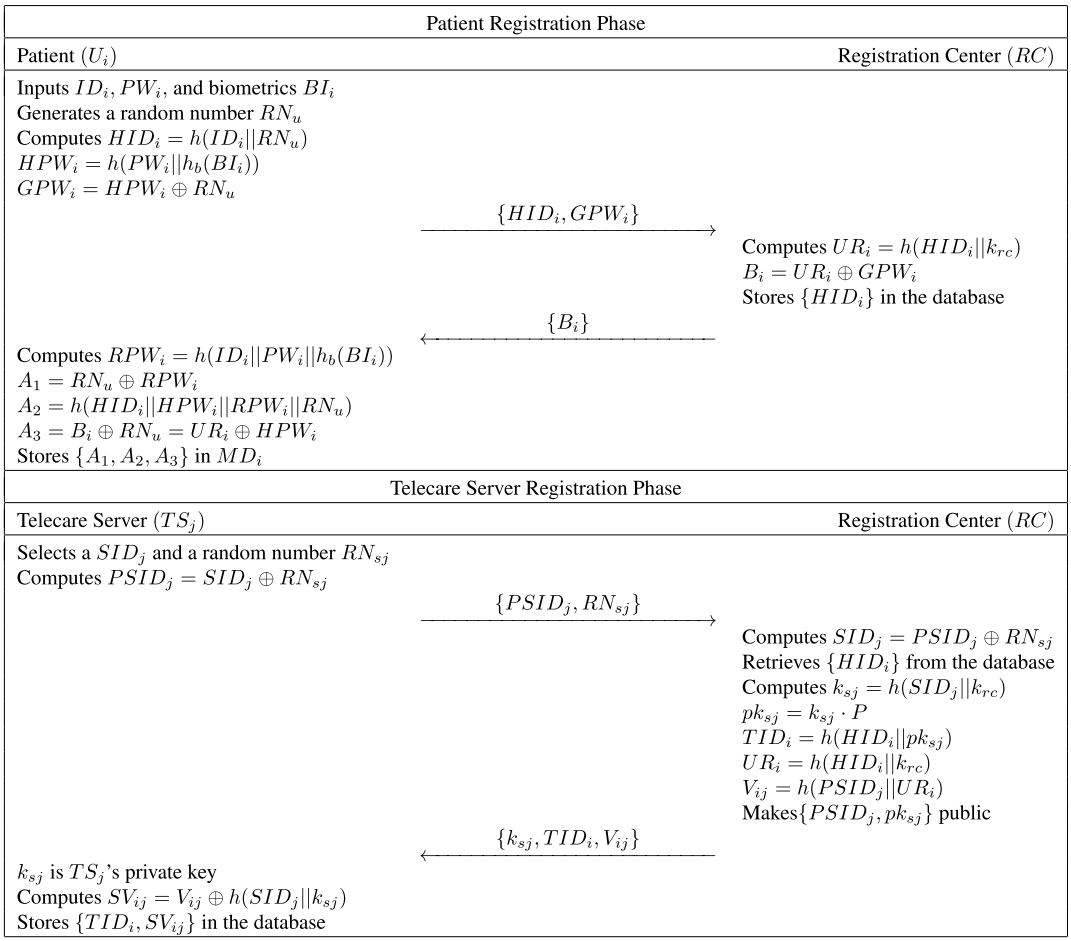

FIGURE 4. Patient and telecare server registration phase of our protocol.

its own database corresponding to TID_i and calculates $V_{ij} = SV_{ij} \oplus h(SID_j||k_{sj})$, $UID_i = h(TID_i||V_{ij}||T_1)$, and $M_1^* = h(UID_i||S_2||V_{ij}||T_1)$. Afterward, TS_j checks whether $M_i^* \stackrel{?}{=} M_i$. If it is the same, TS_j selects a random number n_2 and a timestamp T_2. Finally, TS_j calculates $S_3 = n_2 \cdot P, S_4 = n_2 \cdot S_1, SK = h(UID_i||S_2||S_4), M_j = h(UID_i||SK||T_2)$ and transmits $\{M_j, S_3, T_2\}$ to U_i through a public channel.

2) After receiving $\{M_j, S_3, T_2\}$ from TS_j, U_i verifies the timestamp T_2 by the condition $|T_2^* - T_2| \leq \Delta T$. Then, U_i calculates $S_4 = n_1 \cdot S_3, SK = h(UID_i||S_2||S_4), M_j^* = h(UID_i||SK||T_2)$, and checks the condition $M_j^* \stackrel{?}{=} M_j$. If it is the same, U_i and TS_j have successfully finished the session key agreement and mutual authentication. The detailed process of the login and authentication phase is shown in Figure 5.

E. PASSWORD CHANGE PHASE

U_i performs the following steps to change the old password.

1) U_i inputs ID_i, PW_i, and BI_i in MD_i.
2) MD_i computes $RPW_i = h(ID_i||PW_i||h_b(BI_i))$, $RN_u = A_1 \oplus RPW_i, HID_i = h(ID_i||RN_u)$, $HPW_i = h(PW_i||h_b(BI_i))$, and $A_2^* = h(HID_i||HPW_i||RPW_i||RN_u)$. Then, MD_i checks whether $A_2^* \stackrel{?}{=} A_2$. If it is equal, MD_i requests a new password to U_i.
3) U_i selects a new password PW_i^{new} and sends PW_i^{new} to MD_i.
4) After getting PW_i^{new}, MD_i computes $HPW_i^{new} = h(PW_i^{new}||h_b(BI_i))$, $RPW_i^{new} = h(ID_i||PW_i^{new}||h_b(BI_i))$, $A_1^{new} = RN_u \oplus RPW_i^{new}$, $A_2^{new} = h(HID_i||HPW_i^{new}||RPW_i^{new}||RN_u)$, and $A_3^{new} = UR_i \oplus HPW_i^{new}$. Finally, MD_i replaces $\{A_1, A_2, A_3\}$ with $\{A_1^{new}, A_2^{new}, A_3^{new}\}$.

VII. SECURITY ANALYSIS

In this section, we perform informal and formal security analyses of our protocol using AVISPA, BAN logic, and ROR model. We demonstrate that our protocol can prevent a variety of security attacks.

A. INFORMAL SECURITY ANALYSIS

We conduct the informal security analysis to prove that our protocol provides various security features. Through the analysis, our proposed protocol can prevent a variety of security attacks including stolen mobile devices,

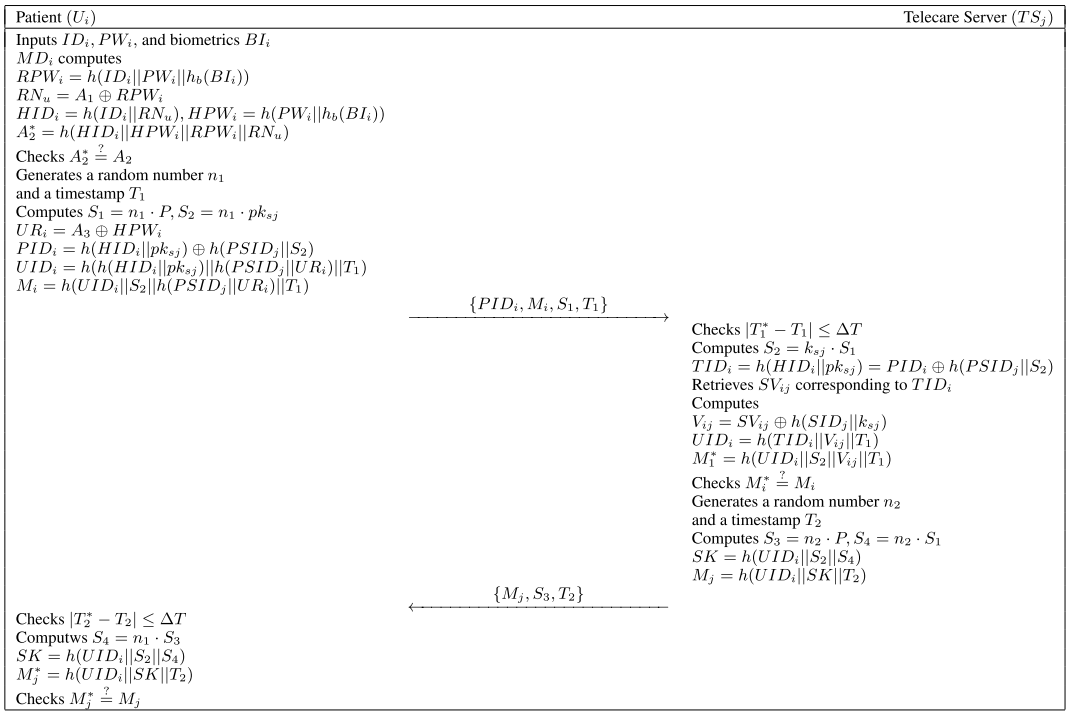

FIGURE 5. Login and authentication phase of our protocol.

insider, and MITM attacks. In addition, our protocol ensures patient anonymity, mutual authentication, and proper password change.

1) STOLEN MOBILE DEVICE ATTACK

We assume that an adversary U_a obtains or steals MD_i. Then, U_a can extract the stored parameters $\{A_1, A_2, A_3\}$ in MD_i. However, U_a cannot obtain any information of U_i because all parameters stored in MD_i are masked with ID_i, PW_i, and BI_i such as $A_1 = RN_u \oplus h(ID_i||PW_i||h_b(BI_i))$. Therefore, our protocol can withstand stolen mobile device attacks.

2) OFFLINE PASSWORD GUESSING ATTACK

If U_a successfully extracts the stored parameters $\{A_1, A_2, A_3\}$ in MD_i and eavesdrop transmitted message $\{PID_i, M_i, S_1, T_1\}$ via an insecure channel, then U_a can attempt to calculate information of U_i. However, U_i's information is masked with ID_i, PW_i, and BI_i such as $RN_u = A_1 \oplus h(ID_i||PW_i||h_b(BI_i))$ and $UR_i = A_3 \oplus h(PW_i||h_b(BI_i))$, so that U_a cannot compute any information of U_i without knowing ID_i and BI_i. Thus, our protocol can prevent offline password guessing attacks.

3) FORGERY ATTACK

If U_a wants to forge U_i's login request message, U_a should create $\{PID_i, M_i, S_1, T_1\}$. However, U_a cannot generate a correct login request message because U_a does not know the parameters TID_i, UR_i, and n_1. Likewise, U_a cannot generate TS_j's response message $\{M_j, S_3, T_2\}$ without knowing UID_i and n_2. Therefore, our protocol can prevent forgery attacks.

4) KEY COMPROMISE IMPERSONATION ATTACK

Assume that U_a can eavesdrop on the login request message and get the private key $\{k_{sj}, k_{rc}\}$. Then, U_a can try to create the login request message $\{PID_i^*, M_i^*, S_1^*, T_1^*\}$ to impersonate a legal patient U_i. However, U_a cannot generate login request message without ID_i, PW_i, BI_i, and random numbers RN_u, n_1. Therefore, our protocol can prevent key compromise impersonation attacks.

5) REPLAY AND MITM ATTACKS

Suppose that U_a intercepts the transmitted messages $\{PID_i, M_i, S_1, T_1\}$ and $\{M_j, S_3, T_2\}$ through a public channel. However, U_a cannot impersonate U_i and TS_j by resending the messages because they verify the random numbers $\{n_1, n_2\}$ and timestamp $\{T_1, T_2\}$. Furthermore, U_a cannot generate valid parameters M_i, S_1, M_j, and S_3 because they are generated by random numbers and timestamps such as $S_1 = n_1 \cdot P$ and $M_i = h(UID_i||S_2||h(PSID_j||UR_i)||T_1)$. As a result, our protocol can withstand replay and MITM attacks.

6) PERFECT FORWARD SECRECY

Suppose that U_a obtains long-term secret keys $\{k_{sj}, k_{rc}\}$ and eavesdrops on the transmitted message $\{PID_i, M_i, S_1, T_1\}$. Then, U_a can attempt to calculate $SK = h(UID_i||S_2||S_4)$. However, U_a cannot compute the parameters UID_i and S_4 without the secret value UR_i, V_{ij}, and random number n_2. Therefore, our protocol guarantees the perfect forward secrecy.

7) PATIENT ANONYMITY

If U_a can eavesdrop on transmitted messages and obtain MD_i, U_a cannot obtain U_i's real identity ID_i. In our protocol, U_i sends the pseudo identity $PID_i = h(HID_i||pk_{sj}) \oplus h(SID_j||S_2)$ to TS_j. However, U_a cannot compute $TID_i = h(HID_i||pk_{sj})$ without the random number n_1 or a private key k_{sj} of TS_j. Furthermore, U_i sends $HID_i = h(ID_i||RN_u)$ instead of ID_i to RC for performing the patient registration phase so that ID_i is not revealed any entities during communication. For these reasons, our protocol provides patient anonymity.

8) TELECARE SERVER SPOOFING ATTACK

Let be U_a intercepts the messages through a public channel and knows TS_j's private key k_{sj} for masquerading TS_j. Then, U_a tries to generate $\{M_j^*, S_3^*, T_2^*\}$ to deceive any legal patient. However, U_a cannot generate the parameter M_j^* without knowing $UID_i = h(TID_i||V_{ij}||T_1)$ because V_{ij} is masked by SV_{ij} that is only stored in TS_j's database. Thus, our protocol can prevent the telecare server spoofing attacks.

9) INSIDER ATTACK

Suppose that U_a registers to RC as a legal patient and intercepts the transmitted messages $\{PID_i, M_i, S_1, T_1\}$ and $\{M_j, S_3, T_2\}$. However, U_a cannot compute vital information such as $UR_i = h(HID_i||k_{rc})$ and $S_2 = n_1 \cdot Pk_{sj}$ to impersonate U_i. Because U_a cannot obtain RC's private key k_{rc} and random number n_1 so that U_a cannot generate legal patient U_i's login request message. As a result, our protocol can protect against insider attacks.

10) PRIVILEGED INSIDER ATTACK

Let be U_a as a privileged insider user of RC. Then, U_a can obtain the registration information $\{HID_i, GPW_i\}$ of U_i. Furthermore, U_a can extract the parameters $\{A_1, A_2, A_3\}$ from MD_i of U_i. However, U_a cannot compute any information of U_i such as RN_u and UR_i without ID_i, PW_i, and BI_i. Thus, our protocol can resist privileged insider attacks.

11) EPHEMERAL SECRET LEAKAGE ATTACK

According to Section III-C, U_a can compromise the short-term (ephemeral) secret and long-term secret parameters. Then, U_a can try to compute $SK = h(UID_i||S_2||S_4)$ established between U_i and TS_j. The two cases are presented below.

- Assume that short-term secret parameters n_1 and n_2 are revealed to U_a. Then, U_a attempts to compute SK. Although U_a can compute S_2 and S_4 with the short-term secret parameters, UID_i cannot be computed without the long-term secret parameters k_{rc} and k_{sj}.
- Assume that long-term secret parameters k_{rc} and k_{sj} are revealed to U_a. However, U_a still cannot compute SK because U_a does not know the short-term secret parameters n_1 and n_2.

Following the above two cases, U_a must be aware of both short-term and long-term secret parameters to generate the correct SK. As a result, our protocol is resistant to ESL attacks.

12) STOLEN VERIFIER ATTACK

Assume that U_a can steal $\{TID_i, SV_{ij}\}$ in TS_j's verification table. Then, U_a intercepts transmitted messages $\{PID_i, M_i, S_1, T_1\}$ and $\{M_j, S_3, T_2\}$ to compute $SK = h(UID_i||S_2||S_4)$. However, U_a cannot compute $UID_i = h(TID_i||V_{ij}||T_1)$ because V_{ij} is masked by $h(SID_j||k_{sj})$. Moreover, U_a cannot obtain $S_2 = n_1 \cdot pk_{sj}$ and $S_4 = n_2 \cdot S_1$ without knowing random numbers n_1 and n_2. Thus, our protocol can withstand stolen verifier attacks.

13) MUTUAL AUTHENTICATION

U_i and TS_j perform the verification processes. After receiving the login request message $\{PID_i, M_i, S_1, T_1\}$ from U_i, TS_j checks $M_i^* \stackrel{?}{=} M_i$. If it is equal, TS_j authenticates U_i. Then, TS_j transmits response message $\{M_j, S_3, T_2\}$ to U_i. Afterward, U_i checks $M_j^* \stackrel{?}{=} M_j$. If it is valid, U_i successfully authenticates TS_j. Therefore, our protocol achieves mutual authentication.

14) SECURE PASSWORD CHANGE

In our protocol, U_i can change the old password PW_i by inputting ID_i, PW_i, and BI_i in MD_i. Then, MD_i receives a new password PW_i^{new} from U_i and computes $RPW_i^{new}, A_1^{new}, A_2^{new}$, and A_3^{new}. Then, MD_i replaces $\{A_1, A_2, A_3\}$ with $\{A_1^{new}, A_2^{new}, A_3^{new}\}$ for future purpose. Therefore, U_a cannot arbitrarily change U_i's password because U_a does not know ID_i, PW_i, and BI_i. Hence, our protocol provides a secure password change.

B. FORMAL SECURITY ANALYSIS USING BAN LOGIC

We perform BAN logic analysis which is generally used to demonstrate secure mutual authentication of the protocol [33]–[35]. Table 2 shows the notations of BAN logic before defining the goals, idealized forms, and assumptions. Following that, we perform the BAN logic proof using the BAN logic rules.

1) BAN LOGIC RULES

The logical rules are shown below.

- Message meaning rule (MMR):

$$\frac{S_1| \equiv S_1 \stackrel{K}{\leftrightarrow} S_2, S_1 \triangleleft \{L_1\}_K}{S_1| \equiv S_2| \sim L_1}$$

- Nonce verification rule (NVR):

$$\frac{S_1| \equiv \#(L_1), S_1| \equiv S_2| \sim L_1}{S_1| \equiv S_2| \equiv L_1}$$

- Jurisdiction rule (JR):

$$\frac{S_1| \equiv S_2 \Rightarrow L_1, S_1| \equiv S_2| \equiv L_1}{S_1| \equiv L_1}$$

TABLE 2. Notaions of BAN logic.

Notaion	Definition
S_1, S_2	Two principals
L_1, L_2	Two statements
SK	Session key
$S_1 \mid \equiv L_1$	S_1 believes L_1
$S_1 \mid \sim L_1$	S_1 once said L_1
$S_1 \triangleleft L_1$	S_1 receives L_1
$S_1 \Rightarrow L_1$	S_1 controls L_1
$\#L_1$	L_1 is fresh
$\{L_1\}_K$	L_1 is encrypted by K
$S_1 \overset{K}{\leftrightarrow} S_2$	S_1 and S_2 communicate via shared key K
$S_1 \overset{L_2}{\rightleftharpoons} S_2$	L_2 is a secret known only to S_1 and S_2

- Belief rule (BR):

$$\frac{S_1 \mid \equiv (L_1, L_2)}{S_1 \mid \equiv L_1}$$

- Freshness rule (FR):

$$\frac{S_1 \mid \equiv \#(L_1)}{S_1 \mid \equiv \#(L_1, L_2)}$$

2) GOALS

The proposed protocol should achieve the following goals to demonstrate secure mutual authentication.

Goal 1: $U_i \mid \equiv (U_i \overset{SK}{\leftrightarrow} TS_j)$

Goal 2: $U_i \mid \equiv TS_j \mid \equiv (U_i \overset{SK}{\leftrightarrow} TS_j)$

Goal 3: $TS_j \mid \equiv (U_i \overset{SK}{\leftrightarrow} TS_j)$

Goal 4: $TS_j \mid \equiv U_i \mid \equiv (U_i \overset{SK}{\leftrightarrow} TS_j)$

3) IDEALIZED FORMS

We can idealize the communicated messages $\{PID_i, M_i, S_1, T_1\}$ and $\{M_j, S_3, T_2\}$ of our protocol as follows.

Message 1:

$$U_i \rightarrow TS_j : \{TID_i, S_1, T_1\}_{S_2}$$

Message 2:

$$TS_j \rightarrow U_i : \{S_3, T_2\}_{S_4}$$

4) ASSUMPTIONS

The assumptions of our proposed protocol are presented as below.

A_1: $TS_j \mid \equiv (U_i \overset{S_2}{\leftrightarrow} TS_j)$

A_2: $TS_j \mid \equiv \#(T_1)$

A_3: $U_i \mid \equiv (U_i \overset{S_4}{\leftrightarrow} TS_j)$

A_4: $U_i \mid \equiv \#(T_2)$

A_5: $TS_j \mid \equiv U_i \Rightarrow (U_i \overset{SK}{\leftrightarrow} TS_j)$

A_6: $U_i \mid \equiv TS_j \Rightarrow (U_i \overset{SK}{\leftrightarrow} TS_j)$

A_7: $TS_j \mid \equiv (U_i \overset{V_{ij}}{\rightleftharpoons} TS_j)$

A_8: $U_i \mid \equiv (U_i \overset{h(PSID_j||UR_i)}{\rightleftharpoons} TS_j)$

5) BAN LOGIC PROOF

We use BAN logic rules and assumptions to prove the mentioned goals. The detailed steps are below.

- S_1 can be obtained from Message 1.

$$S_1 : TS_j \triangleleft \{TID_i, S_1, T_1\}_{S_2}$$

- For obtaining S_2, we apply the MMR with S_1 and A_1.

$$S_2 : TS_j \mid \equiv U_i \mid \sim \{TID_i, S_1, T_1\}$$

- For obtaining S_3 we apply the FR with S_2 and A_2.

$$S_3 : TS_j \mid \equiv \#(TID_i, S_1, T_1)$$

- For obtaining S_4, we apply the NVR with S_2 and S_3.

$$S_4 : TS_j \mid \equiv U_i \mid \equiv (TID_i, S_1, T_1)$$

- S_5 can be obtained from Message 2.

$$S_5 : U_i \triangleleft \{S_3, T_2\}_{S_4}$$

- For obtaining S_6, we apply the MMR with S_5 and A_3.

$$S_6 : U_i \equiv TS_j \mid \sim (S_3, T_2)$$

- For obtaining S_7, we apply the FR with S_6 and A_4.

$$S_7 : U_i \equiv \#(S_3, T_2)$$

- For obtaining S_8, we apply the NVR with S_6 and S_7.

$$S_8 : U_i \mid \equiv TS_j \mid \equiv (S_3, T_2)$$

- For obtaining S_9, we apply the BR with S_8

$$S_9 : U_i \mid \equiv TS_j \mid \equiv (S_3)$$

- S_{10} can be obtained by S_4 and A_7. TS_j can compute $UID_i = h(TID_i||V_{ij}||T_1)$, $S_2 = k_{sj} \cdot S_1$, and the session key $SK = h(UID_i||S_2||S_4)$.

$$S_{10} : TS_j \mid \equiv U_i \mid \equiv (U_i \overset{SK}{\leftrightarrow} TS_j) \quad (Goal\ 4)$$

- S_{11} can be obtained by S_9 and A_8. U_i can compute $UID_i = h(TID_i||h(PSID_j||UR_i)||T_1)$, $S_4 = n_1 \cdot S_3$, and the session key $SK = h(UID_i||S_2||S_4)$.

$$S_{11} : U_i \mid \equiv TS_j \mid \equiv (U_i \overset{SK}{\leftrightarrow} TS_j) \quad (Goal\ 2)$$

- S_{12} can be obtained by applying the JR using S_{10} and A_5.

$$S_{12} : TS_j \mid \equiv (U_i \overset{SK}{\leftrightarrow} TS_j) \quad (Goal\ 3)$$

- S_{13} can be obtained by applying the JR using S_{11} and A_6.

$$S_{13} : U_i \mid \equiv (U_i \overset{SK}{\leftrightarrow} TS_j) \quad (Goal\ 1)$$

Thus, our protocol provides mutual authentication between U_i and TS_j.

C. FORMAL SECURITY ANALYSIS USING ROR MODEL

The ROR model is widely used in various authentication protocols [36]–[38] to demonstrate the security of session keys. We demonstrate that the session key in our protocol is probabilistically safe using the ROR model. We denote P^t as a participant with t instance. Then, we can represent $P_{U_i}^{t_1}$ and $P_{TS_j}^{t_2}$ as participants, where t_1 and t_2 are patient and telecare server instances, respectively. An adversary A can use queries such as *Execute*, *Send*, *CorruptMD*, and *Test* to carry out various security attacks under the ROR model.

- $Execute(P_{U_i}^{t_1}, P_{TS_j}^{t_2})$: A can eavesdrop on the transmitted message between $P_{U_i}^{t_1}$ and $P_{TS_j}^{t_2}$.
- $Send(P^t, Msg)$: A sends the message Msg to P^t and receives the response of Msg.
- $CorruptMD(P_{U_i}^{t_1})$: A can obtain MD_i of $P_{U_i}^{t_1}$ and extract the stored information.
- $Test(P^t)$: A obtains a flipped unbiased coin c before the game starts. If A executes the *Test* query, P^t returns SK when $c = 1$ and a random number when $c = 0$. Otherwise, P^t returns a null ($\perp$).

1) SECURITY PROOF

Theorem 1: Let A be an adversary running in time t against our protocol P and l be the number of bits in the biometrics BI_i. We also indicate that $q_h, q_s, |Hash|, |D_1|, |D_2|$ represent the number of hash queries performed by A, the number of send queries performed by A, the range space of the hash function, the size of the identity dictionary and the size of the password dictionary, respectively. Finally, the probability of breaking *ECDDHP* is given by $Adv_A^{ECDDHP}(t)$. Then, we can obtain the following.

$$Adv_p(t) \leq \frac{q_h^2}{|Hash|} + 2\left(\frac{q_s}{|D_1| \cdot |D_2| \cdot 2^l} + Adv_A^{ECDDHP}(t)\right)$$

Proof: We perform the five games G_k, where $k = 0, 1, 2, 3, 4$. We also define $Succ_j^A$ that A wins G_k by guessing the random bit c. In addition, the probability of A winning game G_k is defined as $Pr_A[Succ_j^A]$. The process of the games is as follows.

- *Game* G_0: A chooses the bit c at the start of G_0. In G_0, A does not perform a query and has no information about SK. Then, we can obtain the following.

$$Adv_P(t) = |2Pr[Succ_0^A] - 1| \tag{1}$$

- *Game* G_1: A performs $Execute(P_{U_i}^{t_1}, P_{TS_j}^{t_2})$ query and eavesdrops transmitted messages $\{PID_i, M_i, S_1, T_1\}$ and $\{M_j, S_3, T_2\}$. Then, A executes *Test* queries to obtain the return value, and A guesses whether the value is the session key SK or a random number. In our protocol, SK is computed as $SK = h(UID_i||S_2||S_4)$. To derive SK, A requires TID_i, n_1, n_2 and these parameters are still unknown to A. Therefore, A's probability of winning G_1 by eavesdropping is not increased.

$$Pr_A[Succ_0^A] = Pr_A[Succ_1^A] \tag{2}$$

- *Game* G_2: To obtain SK, A executes *Hash* and *Send* queries. Also, A uses transmitted messages $\{PID_i, M_i, S_1, T_1\}$ and $\{M_j, S_3, T_2\}$ to obtain SK. However, these messages are protected by random numbers and a hash function. Therefore, A must find the hash collision to win this game because transmitted messages contain no information about SK. According to birthday paradox, we can obtain the following.

$$|Pr_A[Succ_2^A] - Pr_A[Succ_1^A]| \leq \frac{q_h^2}{2|Hash|} \tag{3}$$

- *Game* G_3: A can try to get SK with *CorruptMD*. Then, A can get the information $\{A_1, A_2, A_3\}$, which are $A_1 = RN_u \oplus RPW_i$, $A_2 = h(HID_i||HPW_i||RPW_i||RN_u)$ and, $A_3 = UR_i \oplus HPW_i$. To obtain SK, A needs RN_u and UR_i which is masked with ID_i, PW_i, and BI_i. Therefore, A can try to guess the values from identity dictionary, password dictionary and biometric BI_i of l bits is roughly $1/2^l$. Then, we can obtain the following.

$$|Pr_A[Succ_3^A] - Pr_A[Succ_2^A]| \leq \frac{q_s}{|D_1| \cdot |D_2| \cdot 2^l} \tag{4}$$

- *Game* G_4: In this game, A can try to get $SK = h(UID_i||S_2||S_4)$, use all the eavesdrop messages $\{PID_i, M_i, S_1, T_1\}$ and $\{M_j, S_3, T_2\}$. However, A cannot compute $S_4 = n_1 \cdot S_3 = n_2 \cdot S_1$ even having $S_1 = n_1 \cdot P$ and $S_3 = n_2 \cdot P$ because of the intractability of *ECDDHP*. Then, we can obtain the following.

$$|Pr_A[Succ_4^A] - Pr_A[Succ_3^A]| \leq Adv_A^{ECDDHP}(t) \tag{5}$$

Since all the games are executed, A should guess the bit c through the *Test* query. Then, we can obtain the following.

$$Pr_A[Succ_4^A] = \frac{1}{2} \tag{6}$$

We can obtain the following result using (1), (2), and (6).

$$\begin{aligned} \frac{1}{2}Adv_P(t) &= |Pr_A[Succ_0^A] - \frac{1}{2}| \\ &= |Pr_A[Succ_1^A] - \frac{1}{2}| \\ &= |Pr_A[Succ_1^A] - Pr[Succ_4^A]| \end{aligned} \tag{7}$$

Using the triangular inequality and (3), (4), and (5), we can derive the following result.

$$\begin{aligned} |Pr_A[Succ_1^A] - Pr_A[Succ_4^A]| &\leq |Pr_A[Succ_1^A] - Pr_A[Succ_3^A]| \\ &\quad + |Pr_A[Succ_3^A] - Pr_A[Succ_4^A]| \\ &\leq |Pr_A[Succ_1^A] - Pr_A[Succ_2^A]| \\ &\quad + |Pr_A[Succ_2^A] - Pr_A[Succ_3^A]| \\ &\quad + |Pr_A[Succ_3^A] - Pr_A[Succ_4^A]| \\ &\leq \frac{q_h^2}{2|Hash|} + \frac{q_s}{|D_1| \cdot |D_2| \cdot 2^l} \\ &\quad + Adv_A^{ECDDHP}(t) \end{aligned} \tag{8}$$

Finally, we obtain (9) using (7) and (8).

$$Adv_p(t) \leq \frac{q_h^2}{|Hash|} + 2\left(\frac{q_s}{|D_1| \cdot |D_2| \cdot 2^l} + Adv_A^{ECDDHP}(t)\right) \tag{9}$$

Therefore, we prove Theorem 1.

D. FORMAL SECURITY ANALYSIS USING AVISPA

AVISPA is widely used as a formal security verification tool to verify the security of various protocols that can prevent replay and MITM attacks. Therefore, many researchers used AVISPA to prove the protocol's security [39]–[41]. AVISPA specifies the actions of each type of participant using High-Level Protocols Specifications Language (HLPSL) which is a role-based language. The HLPSL specification of the protocol is translated into Intermediate Format (IF), by a translator called HLPSL2IF. Afterward, IF is input to one of the four backends which produces the back-end output known as Output Format (OF). AVISPA has four back-ends: the On-the-fly-Model-Checker (OFMC), the CL-based Attack Searcher (CL-AtSe), the SAT-based Model-Checker (SATMC), and the Tree-Automata-based Protocol Analyzer (TA4SP). If OF is SAFE, the protocol is resistant to replay and MITM attacks. We utilize OFMC and CL-AtSe backends that provide the XOR operation for AVISPA simulation of our proposed protocol.

1) HLPSL SPECIFICATIONS OF PROPOSED PROTOCOL

We denote three basic roles in HLPSL: the patient U, telecare server TS, and registration center RC. Figure 6 depicts the role of session and environment using the HLPSL language. The role of the session declares all the basic roles and channels used by basic roles. In the role of environment, we declare all used constants and variables and define the intruder knowledge, secrecy goals, and authentication goals. The *secrecy_of* is used to keep secret values and we can check the validity of secret values between participants through the *authentication_on*.

In Figure 7, we present the role of U using the HLPSL language. In transition 1, U starts the registration process with the values in state 0 and updates the state from 0 to 1. Thereafter, U transmits the registration message $\{HID_i, GPW_i\}$ to RC through a secure channel. U receives $\{B_i\}$ from RC in transition 2 and changes the state from 1 to 2. Then, U computes $\{A_1, A_2, A_3\}$ and stores the results in the mobile device. Afterward, U sends request message $\{PID_i, M_i, S_1, T_1\}$ and defines $witness(U, TS, u\_ts\_n_1, N_1)$. In transition 3, U receives the response message from TS. Thereafter, U updates the state from 2 to 3, computes the session key, and defines $request(TS, U, ts\_u\_n_2, N_2)$.

2) RESULT OF AVISPA SIMULATION

Figure 8 depicts the simulation result of our protocol after performing the OFMC and CL-AtSe backends. Following the result, our protocol can prevent replay and MITM attacks because the summary parts are SAFE.

```
%%%% Role Session %%%%
role session(U,TS,RC:agent,SKurc,SKtsrc:symmetric_key,H,Mul:hash_func)
def=
local SN1,SN2,SN3,RV1,RV2,RV3:channel(dy)
composition
patient(U,TS,RC,SKurc,SKtsrc,H,Mul,SN1,RV1)
/\tserver(U,TS,RC,SKurc,SKtsrc,H,Mul,SN2,RV2)
/\rcenter(U,TS,RC,SKurc,SKtsrc,H,Mul,SN3,RV3)
end role

%%%% Role environment %%%%
role environment()
def=
const u,ts,rc:agent,
skurc,sktsrc:symmetric_key,
h,mul:hash_func,
idi,pwi,bioi,mu,hidi,hpwi,gpwi,rpwi,a1,a2,a3,n1,t1,s1,s2,pidi,uidi,mi,uri,
sidj,psidj,ksj,pksj,n2,t2,s3,s4,svij,sk,mj,
krc,tidi,p,vij,bi:text,
sp1,sp2,sp3,sp4,sp5,sp6,
u_ts_n1,ts_u_n2
:protocol_id

intruder_knowledge={u,ts,rc,pidi,psidj,p,pksj,s1,s3,t1,t2,h,mul}
composition
session(u,ts,rc,skurc,sktsrc,h,mul)
/\session(i,ts,rc,skurc,sktsrc,h,mul)
/\session(u,i,rc,skurc,sktsrc,h,mul)
/\session(u,ts,i,skurc,sktsrc,h,mul)
end role

%%% goal %%%%
goal
secrecy_of sp1,sp2,sp3,sp4,sp5,sp6
authentication_on u_ts_n1
authentication_on ts_u_n2
end goal

environment()
```

FIGURE 6. Role of session, environment, and goal.

```
%%%% Patient %%%%
role patient(U,TS,RC:agent,SKurc,SKtsrc:symmetric_key,H,Mul:hash_func,SN,RV:channel(dy))
played_by U
def=
local State:nat,
      IDi,PWi,BIOi,RNu,HIDi,HPWi,GPWi,RPWi,A1,A2,A3,N1,T1,S1,S2,PIDi,UIDi,Mi,URi:text,
      SIDj,PSIDj,RNsj,Ksj,PKsj,N2,T2,S3,S4,SVij,SK,Mj:text,
      Krc,TIDi,P,Vij,Bi:text
const sp1,sp2,sp3,sp4,sp5,sp6,u_ts_n1,ts_u_n2:protocol_id
init State:=0
transition

%%%% Patient registration phase %%%%
1. State=0 /\RV(start)=|>
State':=1 /\RNu':=new()
/\HIDi':=H(IDi.RNu') /\HPWi':=H(PWi.H(BIOi)) /\GPWi':=xor(HPWi',RNu')
/\SN{{HIDi'.GPWi'}_SKurc)
/\secret{{IDi.PWi.BIOi.RNu'},sp1,{U}}
/\secret{{HIDi',GPWi'},sp2,{U,RC}}

2.State=1
/\RV{{xor(H(IDi.RNu').Krc),GPWi'}}_SKurc)=|>
State':=2 /\RPWi':=H(IDi.PWi.H(BIOi))
/\A1':=xor(RNu',RPWi') /\A2':=H(H(IDi.RNu').H(PWi.H(BIOi)).RPWi'.RNu') /\A3':=xor(H(H(IDi.RNu').Krc),H(PWi.H(BIOi)))

%%%% Patient login & authentication phase %%%%
/\N1':=new() /\T1':=new() /\S1':=Mul(N1'.P) /\S2':=Mul(N1'.PKsj)
/\URi':=xor(A3,H(PWi.H(BIOi))) /\PIDi':=xor(H(H(IDi.RNu').PKsj).H(SIDj.S2')) /\UIDi':=H(H(H(IDi.RNu').PKsj).H(PSIDj.URi').T1')
/\Mi':=H(UIDi'.S2'.H(PSIDj.URi').T1')
/\SN(PIDi'.Mi'.S1'.T1')
/\witness(U,TS,u_ts_n1,N1')

3.State=2
/\RV(H(H(H(IDi.RNu').H(PSIDj.H(H(IDi.RNu').Krc)).T1').H(H(H(IDi.RNu').PKsj).H(PSIDj.H(H(IDi.RNu').Krc)).T1').Mul(Ksj'.Mul(N1'.P))
.Mul(N2'.Mul(N1'.P))).T2').Mul(N2'.P).T2')=|>

State':=3
/\S4':=Mul(N1'.Mul(N2'.Mul(N1'.P)))
/\SK':=H(H(H(H(IDi.RNu').PKsj).H(PSIDj.xor(A3,H(PWi.H(BIOi)))).T1').Mul(N1'.PKsj).S4')
/\request(TS,U,ts_u_n2,N2')
end role
```

FIGURE 7. Role of patient.

```
% OFMC
% Version of 2006/02/13
SUMMARY
  SAFE
DETAILS
  BOUNDED_NUMBER_OF_SESSIONS
PROTOCOL
  /home/span/span/testsuite/results/Seok2.if
GOAL
  as_specified
BACKEND
  OFMC
COMMENTS
STATISTICS
  parseTime: 0.00s
  searchTime: 16.08s
  visitedNodes: 1040 nodes
  depth: 9 plies
```

```
SUMMARY
  SAFE

DETAILS
  BOUNDED_NUMBER_OF_SESSIONS
  TYPED_MODEL

PROTOCOL
  /home/span/span/testsuite/results/Seok2.if

GOAL
  As Specified

BACKEND
  CL-AtSe

STATISTICS

  Analysed  : 15 states
  Reachable  : 15 states
  Translation: 0.38 seconds
  Computation: 0.10 seconds
```

FIGURE 8. Simulation result of OFMC and CL-AtSe.

VIII. PERFORMANCE ANALYSIS

In this section, we evaluate the computation costs, communication costs, and security features of our protocol compared with the related existing protocols.

TABLE 3. Computation costs comparison.

Protocols	Computation costs	Total times
Shamshad et al. [3]	$4T_{ecm} + 10T_h + 2T_{sye}$	0.274700 s
Sutrala et al. [12]	$2T_e + 16T_h$	1.052000 s
Ostad-Sharif et al. [15]	$5T_{ecm} + 4T_{eca} + 22T_h + 2T_{sye}$	0.344823 s
Qi and Chen [21]	$6T_{ecm} + 15T_h + 2T_{sye} + T_H$	0.413350 s
Sahoo et al. [22]	$7T_{ecm} + 13T_h + 2T_{sye} + 2T_H$	0.485425 s
Sahoo et al. [4]	$8T_{ecm} + 18T_h + 6T_{sye} + T_{fuz}$	0.628875 s
Ours	$6T_{ecm} + 19T_h + 2T_H$	0.407950 s

TABLE 4. Communication costs comparison.

Protocols	Communication costs	Messages
Shamshad et al. [3]	2112 bits	2
Sutrala et al. [12]	2368 bits	3
Ostad-Sharif et al. [15]	2212 bits	2
Qi and Chen [21]	2240 bits	3
Sahoo et al. [22]	2016 bits	3
Sahoo et al. [4]	3328 bits	4
Ours	1472 bits	2

A. COMPUTATION COSTS

We evaluate the computation costs of our protocol. Following the [15] and [42], we define execution times T_{ecm}, T_{eca}, T_{fuz}, T_e, T_{sye}, T_h, and T_H as the ECC point multiplication (≈0.063075 s), ECC point addition (≈0.000262 s), fuzzy extractor (≈0.063075 s), modular exponentiation (≈0.522 s), symmetric encryption/decryption (≈0.0087 s), hash function (≈0.0005 s) and biohashing function (≈0.01 s), respectively. Table 3 represents the outcome of the computation costs comparison in authentication phase. [3], [12], [15], [21], and [22] are two party authentication and [4] is three party authentication proposed in TMIS environments. [3] and [15] have lower computational costs than our protocol because the registration center is not used in their authentication protocol. If the patient wants to access the services of another telecare server that the patient has not registered, the patient must register with the telecare server. However, our protocol allows a patient to register with the registration center only once to use the services of all telecare servers. Therefore, our protocol offers more convenience and a wider range of security features.

B. COMMUNICATION COSTS

We compare the communication costs of our protocol with previous related protocols. We define the ECC point, SHA-256 hash function output, identity/password, timestamp, symmetric encryption/decryption, RSA encryption/decryption, and random number as 320, 256, 160, 32, 128, 1024, and 160 bits, respectively. During our login and authentication phase, we exchanged messages $\{PID_i, M_i, S_1, T_1\}$ and $\{M_j, S_3, T_2\}$ that require (256+256+320+32) bits and (256 + 320 + 32) bits, respectively. Following that, our protocol's total communication costs are 1472 bits. According to Table 4, our protocol has lower total communication costs than other related protocols.

TABLE 5. Security features.

Security features	[3]	[12]	[15]	[21]	[22]	[4]	Ours
S_1	Yes	Yes	No	No	Yes	Yes	Yes
S_2	Yes	Yes	Yes	Yes	-	Yes	Yes
S_3	Yes	-	-	Yes	-	Yes	Yes
S_4	Yes	Yes	Yes	Yes	Yes	Yes	Yes
S_5	Yes	No	No	No	-	Yes	Yes
S_6	Yes	No	Yes	Yes	Yes	No	Yes
S_7	Yes	Yes	Yes	Yes	Yes	Yes	Yes
S_8	Yes	-	Yes	Yes	Yes	Yes	Yes
S_9	Yes	No	-	-	-	No	Yes
S_{10}	Yes	Yes	Yes	Yes	-	No	Yes
S_{11}	-	Yes	-	Yes	Yes	Yes	Yes
S_{12}	-	Yes	Yes	-	Yes	Yes	Yes

Note: S_1: impersonation attack; S_2: Replay attack; S_3: MITM attack; S_4: Stolen smartcard/device attack; S_5: offline password guessing attack; S_6:user anonymity; S_7:mutual authentication; S_8:perfect forward secrecy; S_9:insider attack; S_{10}:privileged insider attack; S_{11}:formal security verification using BAN logic; S_{12}:formal security verificiation using AVISPA.

C. SECURITUY FEATURES

Table 5 compares the security features of the proposed protocol to those of previous related protocols. According to Table 5, our protocol can prevent additional security attacks including impersonation, replay, MITM, and insider attacks. As a result, our protocol has more security features than related existing protocols.

IX. CONCLUSION

In this paper, we proved that Sahoo *et al.*'s protocol is vulnerable to insider and privileged insider attacks. In addition, we showed that their protocol cannot guarantee patient anonymity and correct password change. To address the security flaws of their protocol, we proposed a secure ECC-based mutual authentication protocol for TMIS environments. We conducted the informal analysis to demonstrate that our protocol can prevent a variety of security attacks, including stolen mobile device, insider, and privileged insider attacks. Furthermore, we demonstrated that our protocol can ensure mutual authentication and session key security using the BAN logic and the ROR model. We also used the AVISPA to demonstrate that our protocol can withstand replay and MITM attacks. Finally, we conducted a performance analysis on our protocol. Following the results, our protocol provides lower communication costs and better security than related existing protocols. Therefore, our protocol is suitable for the TMIS environments.

REFERENCES

[1] S. Son, J. Lee, M. Kim, S. Yu, A. K. Das, and Y. Park, "Design of secure authentication protocol for cloud-assisted telecare medical information system using blockchain," *IEEE Access*, vol. 8, pp. 192177–192191, 2020.
[2] C.-L. Hsu, T.-V. Le, M.-C. Hsieh, K.-Y. Tsai, C.-F. Lu, and T.-W. Lin, "Three-factor UCSSO scheme with fast authentication and privacy protection for telecare medicine information systems," *IEEE Access*, vol. 8, pp. 196553–196566, 2020.
[3] S. Shamshad, M. F. Ayub, K. Mahmood, S. Kumari, S. A. Chaudhry, and C.-M. Chen, "An enhanced scheme for mutual authentication for healthcare services," *Digit. Commun. Netw.*, Jul. 2021.
[4] S. S. Sahoo, S. Mohanty, and B. Majhi, "A secure three factor based authentication scheme for health care systems using IoT enabled devices," *J. Ambient Intell. Humanized Comput.*, vol. 12, no. 1, pp. 1419–1434, Jan. 2021.

[5] M. Burrows, M. Abadi, and R. Needham, "A logic of authentication," *ACM Trans. Comput. Syst.*, vol. 8, no. 1, pp. 18–36, 1990.

[6] M. Abdalla, P. Fouque, and D. Pointcheval, "Password-based authenticated key exchange in the three-party setting," in *Proc. 8th Int. Workshop Theory Pract. Public Key Cryptogr. (PKC)*, Jan. 2005, pp. 65–84.

[7] AVISPA. *Automated Validation of Internet Security Protocols and Applications*. Accessed: Nov. 2021. [Online]. Available: http://www.avispa-project.org/

[8] SPAN. emphA Security Protocol Animator for AVISPA. Accessed: Nov. 2021. [Online]. Available: http://www.avispa-project.org/

[9] M. K. Khan and S. Kumari, "An authentication scheme for secure access to healthcare services," *J. Med. Syst.*, vol. 37, no. 4, p. 9954, Aug. 2013.

[10] D. Giri, T. Maitra, R. Amin, and P. D. Srivastava, "An efficient and robust RSA-based remote user authentication for telecare medical information systems," *J. Med. Syst.*, vol. 39, no. 1, pp. 1–9, Jan. 2015.

[11] R. Amin and G. P. Biswas, "An improved RSA based user authentication and session key agreement protocol usable in TMIS," *J. Med. Syst.*, vol. 39, no. 8, pp. 1–14, Aug. 2015.

[12] A. K. Sutrala, A. K. Das, V. Odelu, M. Wazid, and S. Kumari, "Secure anonymity-preserving password-based user authentication and session key agreement scheme for telecare medicine information systems," *Comput. Methods Programs Biomed.*, vol. 135, pp. 167–185, Oct. 2016.

[13] L. Zhang and S. Zhu, "Robust ECC-based authenticated key agreement scheme with privacy protection for telecare medicine information systems," *J. Med. Syst.*, vol. 39, no. 5, pp. 1–11, May 2015.

[14] W. Liu, Q. Xie, S. Wang, and B. Hu, "An improved authenticated key agreement protocol for telecare medicine information system," *SpringerPlus*, vol. 5, no. 1, pp. 1–16, Dec. 2016.

[15] A. Ostad-Sharif, D. Abbasinezhad-Mood, and M. Nikooghadam, "A robust and efficient ECC-based mutual authentication and session key generation scheme for healthcare applications," *J. Med. Syst.*, vol. 43, no. 1, pp. 1–22, Jan. 2019.

[16] S. Kumari, P. Chaudhary, C.-M. Chen, and M. K. Khan, "Questioning key compromise attack on Ostad–Sharif et al.'s authentication and session key generation scheme for healthcare applications," *IEEE Access*, vol. 7, pp. 39717–39720, 2019.

[17] Y. Lu, L. Li, H. Peng, and Y. Yang, "An enhanced biometric-based authentication scheme for telecare medicine information systems using elliptic curve cryptosystem," *J. Med. Syst.*, vol. 39, no. 3, pp. 1–8, Mar. 2015.

[18] Q. Jiang, Z. Chen, B. Li, J. Shen, L. Yang, and J. Ma, "Security analysis and improvement of bio-hashing based three-factor authentication scheme for telecare medical information systems," *J. Ambient Intell. Humanized Comput.*, vol. 9, no. 4, pp. 1061–1073, 2018.

[19] N. Ravanbakhsh and M. Nazari, "An efficient improvement remote user mutual authentication and session key agreement scheme for E-health care systems," *Multimedia Tools Appl.*, vol. 77, no. 1, pp. 55–88, 2018.

[20] A. Ostad-Sharif, D. Abbasinezhad-Mood, and M. Nikooghadam, "An enhanced anonymous and unlinkable user authentication and key agreement protocol for TMIS by utilization of ECC," *Int. J. Commun. Syst.*, vol. 32, no. 5, Mar. 2019, Art. no. e3913.

[21] M. Qi and J. Chen, "New robust biometrics-based mutual authentication scheme with key agreement using elliptic curve cryptography," *Multimedia Tools Appl.*, vol. 77, no. 18, pp. 23335–23351, 2018.

[22] S. S. Sahoo, S. Mohanty, and B. Majhi, "Improved biometric-based mutual authentication and key agreement scheme using ECC," *Wireless Pers. Commun.*, vol. 111, no. 2, pp. 991–1017, Mar. 2020.

[23] R. Amin, S. H. Islam, P. Gope, K.-K. R. Choo, and N. Tapas, "Anonymity preserving and lightweight multimedical server authentication protocol for telecare medical information system," *IEEE J. Biomed. Health Inform.*, vol. 23, no. 4, pp. 1749–1759, Jul. 2019.

[24] N. Koblitz, "Elliptic curve cryptosystems," *Math. Comput.*, vol. 48, no. 177, pp. 203–209, 1987.

[25] A. T. B. Jin, D. N. C. Ling, and A. Goh, "BioHashing: Two factor authentication featuring fingerprint data and tokenised random number," *Pattern Recognit.*, vol. 37, no. 11, pp. 2245–2255, Apr. 2004.

[26] R. Lumini and L. Nanni, "An improved BioHashing for human authentication," *Pattern Recognit.*, vol. 40, no. 3, pp. 1057–1065, Mar. 2007.

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

[28] D. Kwon, Y. Park, and Y. Park, "Provably secure three-factor-based mutual authentication scheme with PUF for wireless medical sensor networks," *Sensors*, vol. 21, no. 18, p. 6039, Sep. 2021.

[29] P. Kocher, J. Jaffe, and B. Jun, "Differential power analysis," in *Advances in Cryptology* (Lecture Notes in Computer Science), vol. 1666. Berlin, Germany: Springer, Aug. 1999, pp. 388–397.

[30] J. Oh, S. Yu, J. Lee, S. Son, M. Kim, and Y. Park, "A secure and lightweight authentication protocol for IoT-based smart homes," *Sensors*, vol. 21, no. 4, p. 1488, Feb. 2021.

[31] S. Yu, N. Jho, and Y. Park, "Lightweight three-factor-based privacy-preserving authentication scheme for IoT-enabled smart Homes," *IEEE Access*, vol. 9, pp. 126186–126197, 2021.

[32] R. Canetti and H. Krawczyk, "Universally composable notions of key exchange and secure channels," in *Proc. Int. Conf. Theory Appl. Cryptograph. Techn.*, Amsterdam, The Netherlands, 2002, pp. 337–351.

[33] S. Son, Y. Park, and Y. Park, "A secure, lightweight, and anonymous user authentication protocol for IoT environments," *Sustainability*, vol. 13, no. 16, p. 9241, Aug. 2021.

[34] J. Lee, S. Yu, K. Park, Y. Park, and Y. Park, "Secure three-factor authentication protocol for multi-gateway IoT environments," *Sensors*, vol. 19, no. 10, p. 2358, May 2019.

[35] B. A. Alzahrani, A. Irshad, A. Albeshri, and K. Alsubhi, "A provably secure and lightweight patient-healthcare authentication protocol in wireless body area networks," *Wireless Pers. Commun.*, vol. 117, no. 1, pp. 47–69, Mar. 2021.

[36] J. Lee, G. Kim, A. K. Das, and Y. Park, "Secure and efficient honey list-based authentication protocol for vehicular ad hoc networks," *IEEE Trans. Netw. Sci. Eng.*, vol. 8, no. 3, pp. 2412–2425, Jul. 2021.

[37] S. A. Chaudhry, A. Irshad, J. Nebhen, A. K. Bashir, N. Moustafa, Y. D. Al-Otaibi, and Y. B. Zikria, "An anonymous device to device access control based on secure certificate for Internet of Medical Things systems," *Sustain. Cities Soc.*, vol. 75, Dec. 2021, Art. no. 103322.

[38] T.-Y. Wu, L. Yang, Z. Lee, C.-M. Chen, J.-S. Pan, and S. H. Islam, "Improved ECC-based three-factor multiserver authentication scheme," *Secur. Commun. Netw.*, vol. 2021, pp. 1–14, Jan. 2021.

[39] D. K. Kwon, S. J. Yu, J. Y. Lee, S. H. Son, and Y. H. Park, "WSN-SLAP: Secure and lightweight mutual authentication protocol for wireless sensor networks," *Sensors*, vol. 21, no. 3, p. 936, Jan. 2021.

[40] S. Banerjee, A. K. Das, S. Chattopadhyay, S. S. Jamal, J. J. P. C. Rodrigues, and Y. Park, "Lightweight failover authentication mechanism for IoT-based fog computing environment," *Electronics*, vol. 10, no. 12, p. 1417, Jun. 2021.

[41] M. Kim, J. Lee, K. Park, Y. Park, K. H. Park, and Y. Park, "Design of secure decentralized car-sharing system using blockchain," *IEEE Access*, vol. 9, pp. 54796–54810, 2021.

[42] A. Irshad, H. Naqvi, S. A. Chaudhary, M. Usman, M. Shafiq, O. Mir, and A. Kanwal, "Cryptanalysis and improvement of a multi-server authenticated key agreement by Chen and Lee's scheme," *Inf. Technol. Control*, vol. 47, no. 3, pp. 431–446, 2018.

JONGSEOK RYU received the B.S. degree in software from Kyungpook National University, Sangju, South Korea, in 2021. He is currently pursuing the M.S. degree with the Department of Electronic and Electrical Engineering, Kyungpook National University, Daegu, South Korea. His research interests include authentication, cryptography, and communication security.

JIHYEON OH received the B.S. degree in electronics engineering from Kyungpook National University, Daegu, South Korea, in 2020, where she is currently pursuing the M.S. degree with the School of Electronics and Electrical Engineering. Her research interests include authentication, the Internet of Things, and information security.

DEOKKYU KWON received the B.S. degree in electronics engineering from Kyungpook National University, Daegu, South Korea, in 2020, where he is currently pursuing the M.S. degree with the School of Electronic and Electrical Engineering. His research interests include internet of drones, wireless sensor networks, mutual authentication, and information security.

SEUNGHWAN SON received the B.S. degree in mathematics from Kyungpook National University, Daegu, South Korea, in 2019, and the M.S. degree from the School of Electronic and Electrical Engineering. His research interests include authentication, blockchain, cryptography, and information security.

JOONYOUNG LEE received the B.S. and M.S. degrees in electronics engineering from Kyungpook National University, Daegu, South Korea, in 2018 and 2020, respectively, where he is currently pursuing the Ph.D. degree with the School of Electronic and Electrical Engineering. His research interests include authentication, the Internet of Things, and information security.

YOHAN PARK received the B.S., M.S., and Ph.D. degrees in electronic engineering from Kyungpook National University, Daegu, South Korea, in 2006, 2008, and 2013, respectively. He is currently an Assistant Professor with the Department of Computer Engineering, College of Engineering, Keimyung University, Daegu. His research interests include computer networks, mobile security, blockchain, and the IoT.

YOUNGHO PARK (Member, IEEE) received the B.S., M.S., and Ph.D. degrees in electronic engineering from Kyungpook National University, Daegu, South Korea, in 1989, 1991, and 1995, respectively. From 1996 to 2008, he was a Professor with the School of Electronics and Electrical Engineering, Sangju National University, South Korea. From 2003 to 2004, he was a Visiting Scholar with the School of Electrical Engineering and Computer Science, Oregon State University, USA. He is currently a Professor with the School of Electronic and Electrical Engineering, Kyungpook National University. His research interests include computer networks, multimedia, and information security.