Authentication and Secure Key Management in E-Health Services: A Robust and Efficient Protocol Using Biometrics

Article *in* IEEE Access · August 2019

DOI: 10.1109/ACCESS.2019.2935313

CITATIONS
15

READS
139

4 authors, including:

Anwar Ghani
International Islamic University, Islamabad
62 PUBLICATIONS **667** CITATIONS

SEE PROFILE

IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received July 3, 2019, accepted July 26, 2019, date of publication August 14, 2019, date of current version August 28, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2935313

Authentication and Secure Key Management in E-Health Services: A Robust and Efficient Protocol Using Biometrics

ZAHID MEHMOOD[1], ANWAR GHANI[1], GONGLIANG CHEN[2], AND AHMED S. ALGHAMDI[3]

[1]Department of Computer Science and Software Engineering International Islamic University Islamabad, Islamabad 44000, Pakistan
[2]School of Electronic Information and Electrical Engineering, Shanghai Jiao Tong University, Shanghai 200240, China
[3]Information System Department, Faculty of Computing, University of Jeddah, Jeddah 23890, Saudi Arabia

Corresponding author: Anwar Ghani (anwar.ghani@iiu.edu.pk)

ABSTRACT The use of modern technology for the Goodwell of human beings especially in medical science is a hot research area. Telecare Medicine Information System (TMIS) is very popular in health care services in developed countries where a physician can remotely get patients related information. The security of such information is very critical as its misuse can have adverse effects on the patients' life. The information transmitted over a public channel is protected using authentication protocols. For this purpose, various biometrics-based authentication protocols including Omid et al.'s protocol have been proposed. However, in this article, it has been analyzed that Omid et al.'s protocol is susceptible to user impersonation attack and also fails to protect user identity. Hence, to remedy the problems an improved mechanism is needed to secure the three-factor authentication framework for the practical application. Therefore, a robust and efficient biometrics-based authentication and key agreement protocols for E-Health Services has been proposed. Further, it has been shown through formal and informal analysis that the proposed scheme is provably secure.

INDEX TERMS Authentication protocol, biometric authentication, information security, TMIS.

I. INTRODUCTION

Due to the recent advancement in telecommunication technology, its usage is at the peak in business as well as services industries like healthcare services. The extensive use of inexpensive mobile devices, make it easy to provide different services including healthcare services at the doorsteps. TMISs provide appropriate telecare services to different users at home. TMIS also allow doctors to remotely check up the patient's current condition. Since TMIS is used to share critical user-related information, therefore, securing is of critical importance.

In order to protect the patients' secrecy using secure authentication protocol between a server and patients/doctors, many scholars proposed three-factor user authentication schemes [1]–[7]. Previously introduced password and smartcard-based schemes were used for user authentication due to smart card dominance. However, information can be retrieved from the smart card by an adversary as indicated by Witteman [8] and Messerges *et al.* [9]. So, the security of many authentication protocols has been exposed [10]–[12].

Wu *et al.* [13] in 2012, introduced an authentication protocol for TMIS using the smart-card. However, Debiao *et al.* [14] found that the protocol in [13] is exposed to insider and impersonation attacks if the smartcard of the user is either lost or stolen. Furthermore, they introduced an improved version of the scheme to remedy the security loopholes of Wu et al.'s protocol. Later on, Wei *et al.* [15] proved that if the smart-card is lost/stolen both He *et al.* [14] and Wu *et al.* [13] protocols are susceptible to offline password guessing attack if an adversary successfully extract the information from the smart-card. Therefore, a new protocol was proposed in [15] to address the weaknesses of both protocols. However, Zhu [16] argued that Wei *et al.* [15] proposal itself is exposed to offline password guessing attack where they proposed a new and improved scheme.

Considering the limitation of password-based authentication protocols using smart-card [13], [17]–[20] many scholars presented three-factor authentication schemes [21]–[26]. The first proposal to discuss presented by Tan [23] where

The associate editor coordinating the review of this article and approving it for publication was Junggab Son.

TABLE 1. Notation guide.

Notations	Description
$\mathbb{S}, \mathbb{U}_i$	Server and User
id_{ui}, pw_{ui}	User Identity and Password
r_{u1}, r_{s1}	Two Arbitrary Numbers
B_i	User's Biometric
x_s	Server Private Key
$h(.)$	One-way hash functions
$\|\|$	Concatenation operation
$\oplus$	XOR operation
$\mathbb{A}$	The Adversary
$\mathbb{S}_c$	Smart Card

the authors introduced a biometric-based three factor authentication protocol for TMIS. However, it was analyzed by Yan *et al.* [27] proving that this proposal is unable to resist DoS attack. The authors in [27] came up with their own proposal to remedy the problem in [23]. However, recently a new proposal came up from Omid and Nikooghadam [28] proving the exposure of Yan *et al.* [27] not only to impersonation, offline password guessing attacks but also non provisioning of forward secrecy. To address the weaknesses in protocol proposed in [27], Omid et al. came up with a new proposal for secure authentication. However, after a detailed analysis of the protocol presented by Omid and Nikooghadam [28] it has been found that their proposal, in case of the lose of smartcard, is insecure against user impersonation attack. Moreover, the Omid et al. protocol is unable to secure user anonymity. Therefore, a robust and efficient scheme to counter the indicated flaw in Omid et al.'s scheme has been presented in this article. From now on the Omid et al. scheme will be referred to as the baseline scheme/protocol. Contribution of this article as list as follows:

- To perform detailed cryptanalysis of Omid et al. protocol to find out its security loopholes and weaknesses.
- To proposed an improved, robust and efficient authentication protocol which is resistant to various possible attacks.
- To verify the proposed protocol and the strength of its security using an automated tool.
- To perform an analysis of computation and communication cost has been performed to assess the computation and communication efficiency of the proposed protocol.
- To comparatively analyze the proposed protocol with existing state-of-the-art protocols to validate its performance. The comparison is based on security requirements, computation, and communication efficiency.

II. SYMMETRIC CRYPTOGRAPHY PRIMITIVES

This section details the basic concepts and strengths of Symmetric encryption/decryption and hash functions

A. SYMMETRIC ENCRYPTION

The encryption based of symmetric cryptography can be defined by an algorithm $SEN(..)$ with some key $k \in \{0,1\}^*$ and real time **message** $M \in \{0,1\}^*$ and results a corresponding **cipher** $C \in \{0,1\}^* \cup \{\perp\}$. Formally, $C = SEN(k, M)$ as an instance of execution of SEN with inputs k and M, whereas, SEN outputs C after execution.

B. SYMMETRIC DECRYPTION

The decryption can be defined by an algorithm SED with similar key $k \in \{0,1\}^*$ used in SEN and **cipher** $C \in \{0,1\}^*$. SED results a corresponding real time **message** $M \in \{0,1\}^* \cup \{\perp\}$. Formally, $M = SED(k, C)$ as an instance of execution of SED with inputs k and C, whereas, SED outputs M after execution.

The secure encryption/decryption algorithm based on symmetric cryptography qualifies following properties:

- Given $C = SEN(k, M)$, SED and SEN, it is computationally infeasible to compute M without knowledge of k. This property is called Confidentiality.
- Given M, C, SED and SEN, it is computationally infeasible to extract k This property is called resistance to known plain text and known cipher text.

C. HASH FUNCTIONS

A has function $H : \{0,1\}^* \rightarrow Z_q^*$ takes arbitrary length message M and generates a fixed length code $H_f = H(M)$, where H_f is called hash-code or hash-value. A slight change in M brings significant change in output (Avalanche effect). A hash function should posses following properties to qualify as secure:

- Computing hash-code $H_f = H(M)$ is computationally effortless.
- Given $H_f = H(M)$, computing M is computationally infeasible.
- Finding two messages M_1 and M_2 such that $H(M_1) = H(M_2)$ is computationally infeasible. This property is termed as collision resistance.

III. REVIEW OF THE BASELINE PROTOCOL

A detailed review of the baseline protocol has been presented in this section. The protocol comprises of four phases: Registration phase, Login phase, Key agreement phase, Password and biometric change phase. Registration and Key agreement phases are explained in Fig. 1.

A. REGISTRATION PHASE

1) Any user $\mathbb{U}_i$ wants to initiate communication, selects a distinct identity id_{ui}, password pw_{ui} along with N_i (random number). Then after scanning the biometric B_i, the user calculates $pwd_i = h(id_{ui}\|pw_{ui}\|N_i\|B_i)$. Now $\mathbb{U}_i$ using a secure channel, sends $\{id_{ui}, pwd_i\}$ information to the server $\mathbb{S}$.
2) When the registration request is received, the server $\mathbb{S}$ generates a random number b and computes $X_i = h(id_{ui}\|x_s)$,$Y_i = X_i \oplus pwd_i$, $m_i = h(pwd_i\|X_i\|id_{ui})$ and $C_i = h(b\|x_s) \oplus pwd_i$. Now $\mathbb{S}$ stores the values

$User\ \mathbb{U}_i$		$Server\ \mathbb{S}$
Registration		
Generate N_i		
Computes $pwd_i = h(id_{ui}\|\|pw_{ui}\|\|N_i\|\|B_i)$		
	$\xrightarrow{\{id_{ui}, pwd_i\}}$	
		Computes
		$X_i = h(id_{ui\|\|x_s})$
		$Y_i = X_i \oplus pwd_i$
		$m_i = h(pwd_i\|\|X_i\|\|id_{ui})$
		Generates $b \in Z_n^*$
		$C_i = h(b\|\|x_s) \oplus pwd_i$
		Smart Card $\{Y_i, C_i, m_i, h(.)\}$
	$\xleftarrow{\{Smart\ Card\}}$	
Compute		
$g_i = B_i \oplus h(id_{ui}\|\|pw_{ui})$		
$E_i = N_i \oplus h(id_{ui}\|\|pw_{ui})$		
Smart Card $\{Y_i, g_i, E_i, C_i, m_i, h(.)\}$		

$User\ \mathbb{U}_i$		$Server\ \mathbb{S}$
Authentication		
Enter id_{ui}, pw_{ui}, B_i^*		
$B_i = G_i \oplus h(id_{ui}\|\|pw_{ui})$		
$d(B_i, B_i^*) \leq \tau$		
$N_i = E_i \oplus h(id_{ui}\|\|pw_{ui})$		
$pwd_i = h(id_{ui}\|\|pw_{ui}\|\|N_i\|\|B_i)$		
$X_i = Y_i \oplus pwd_i$		
$m_i^{'} = h(pwd_i\|\|X_i\|\|id_{ui})$ and checks $m_i^{'} = m_i$		
Generates $r_{u1} \in Z_n^*$		
$D = pwd_i \oplus C_i = h(b\|\|x_s)$		
$NID_i = id_{ui} \oplus h(D)$		
$G_i = r_{u1} \oplus h(X_i\|\|id_{ui}\|\|D)$		
$H_1 = h(id_{ui}\|\|X_i\|\|D\|\|r_{u1}\|\|t_1)$		
	$\xrightarrow{\{(NID_i, H_1, G_i, t_1)\}}$	
		$t_2 - t_1 > \Delta t$
		$id_{ui}^{'} = NID_i \oplus h(D)$
		$X_i^{'} = h(id_{ui}^{'}\|\|x_s)$
		$r_{u1} = G_i \oplus h(id_{ui}^{'}\|\|X_i\|\|D)$
		$H_1^{'} \stackrel{?}{=} h(id_{ui}^{'}\|\|X^{'}\|\|D\|\|r_{u1}\|\|t_1)$
		Generate $r_{s1} \in Z_n^*$
		$m_2 = r_{s1} \oplus h(id_{ui}\|\|X_i\|\|D)$
		$H_2^{'} = h(r_{s1}\|\|X_i\|\|D\|\|id_{ui}^{'}\|\|t_3)$
	$\xleftarrow{\{(m_2, H_2, t_3)\}}$	
$t_4 - t_3 > \Delta t$		
$r_{s1}^{'} = m_2 \oplus h(id_{ui}\|\|X_i\|\|D)$		
$H_2^{'} = h(r_{s1}^{'}\|\|X_i\|\|D\|\|id_{ui}\|\|t_3) \stackrel{?}{=} H_2$		
$sk = h(X_i\|\|D\|\|id_{ui}\|\|r_{s1}\|\|r_{u1})$		
$H_3 = h(sk)$		
	$\xrightarrow{\{H_3)\}}$	
		$sk = h(X_i\|\|D\|\|id_{ui}^{'}\|\|r_{s1}\|\|r_{u1})$
		$H_3^{'} \stackrel{?}{=} h(sk) = H_3$
$\longleftarrow$	$\boxed{sk = sk_{ui} = sk_s}$	$\longrightarrow$

FIGURE 1. Omid scheme.

$\{Y_i, C_i, m_i\}$ along with $h(.)$ into the $\mathbb{S}_c$ and the using secure channel sends it to the user.

3) On getting the $\mathbb{S}_c$, $\mathbb{U}_i$ computes $g_i = B_i \oplus h(id_{ui}\|pw_{ui})$, $E_i = N_i \oplus h(id_{ui}\|pw_{ui})$ and enters these parameters into $\mathbb{S}_c$. Now the $\mathbb{S}_c$ contains the information $\{Y_i, g_i, E_i, C_i, m_i, h(.)\}$.

B. LOGIN PHASE

When the registration phase concludes, the user $\mathbb{U}_i$ needs to execute the following steps to Login to the server $\mathbb{S}$.

1: $\mathbb{U}_i$ inserts his/her $\mathbb{S}_c$ into card reader, enters his/her identity id_{ui} and password pw_{ui} and scan his/her biometric B_i^*.

Now, the $\mathbb{S}_c$ computes:

$$B_i = g_i \oplus h(id_{ui}\|pw_{ui}) \quad (1)$$

$$checks\ B_i \stackrel{?}{=} B_i^* \quad (2)$$

$$N_i = E_i \oplus h(id_{ui}\|pw_{ui}) \quad (3)$$

$$pwd_i = h(id_{ui}\|pw_{ui}\|N_i\|B_i) \quad (4)$$

$$X_i = Y_i \oplus pwd_i \quad (5)$$

$$Verifies\ m_i^{'} = h(pwd_i\|X_i\|id_{ui}) = m_i \quad (6)$$

$$Generates\ r_{u1} \in Z_n^* \quad (7)$$

$$D = pwd_i \oplus C_i = h(b\|x_s) \quad (8)$$

$$NID_i = id_{ui} \oplus h(D) \quad (9)$$

$$G_i = r_{u1} \oplus h(X_i\|id_{ui}\|D) \quad (10)$$

$$H_1 = h(id_{ui}\|X_i\|D\|r_{u1}\|t_1) \quad (11)$$

Now $\mathbb{U}_i$ transmits the request message $\{NID_i, H_1, G_i, t_1\}$ to $\mathbb{S}$.

C. KEY AGREEMENT PHASE

2: At the arrival of the request message from the user $\mathbb{U}_i$, the server $\mathbb{S}$ verifies validity of its time stamp. If $t_2 - t_1 > \Delta t$, then it is a valid time stamp then $\mathbb{S}$ computes:

$$id_{ui}^{'} = NID_i \oplus h(D) \quad (12)$$

$$X_i^{'} = h(id_{ui}^{'}\|x_s) \quad (13)$$

$$r_{u1} = G_i \oplus h(id_{ui}^{'}\|X_i\|D) \quad (14)$$

Now verifies $H_1' \stackrel{?}{=} h(id_{ui}'\|X^{'}\|D\|r_{u1}\|t_1) = H_1$, in case of failure the server $\mathbb{S}$ terminates the session otherwise, $\mathbb{S}$ generates a random number $r_{s1} \in Z_n^*$ and calculates the following:

$$m_2 = r_{s1} \oplus h(id_{ui}\|X_i\|D) \quad (15)$$

$$H_2 = h(r_{s1}\|X_i\|D\|id_{ui}^{'}\|t_3) \quad (16)$$

After computation $\mathbb{S}$ sends the message $\{m_2, H_2, t_3\}$ to user $\mathbb{U}_i$.

3: At the arrival of the message, the user $\mathbb{U}_i$ verifies its freshness using the time stamp $t_4 - t_3 > \Delta t$ and computes:

$$r_{s1}^{'} = m_2 \oplus h(id_{ui}\|X_i\|D) \quad (17)$$

$$verifies\ H_2^{'} = h(r_{s1}^{'}\|X_i\|D\|id_{ui}\|t_3) = H_2 \quad (18)$$

$$sk_{ui} = h(X_i\|D\|id_{ui}\|r_{s1}\|r_{u1}) \quad (19)$$

$$H_3 = h(sk) \quad (20)$$

Now $\mathbb{U}_i$ transmits $\{H_3\}$ to server $\mathbb{S}$.

4: On receiving the message $\mathbb{S}$, computes the shared session key $sk_s = h(X_i\|D\|id_{ui}^{'}\|r_{s1}\|r_{u1})$ and verifies $H_3^{'} = h(sk)$, if it holds, the session key $sk = sk_{ui} = sk_s$ is considered as valid key between user $\mathbb{U}_i$ and server $\mathbb{S}$.

D. PASSWORD CHANGE PHASE

In case the password or biometric needs to be updated, the user $\mathbb{U}_i$ is required to execute the following steps in sequence:

1) The user $\mathbb{U}_i$ uses a card reader to read information from the smart card. At the same time the user also enters his/her identity id_{ui} and password pw_{ui} and scan biometric. Now the smart-card computes $B_i = g_i \oplus h(id_{ui}\|pw_{ui})$ and verifies if $d(B_i, B_i^*) \leq \tau$. In case the condition holds, the smart-card computes $N_i = E_i \oplus h(id_{ui}\|pw_{ui})$, $pwd_i = h(id_{ui}\|pw_{ui}\|N_i\|B_i)$, $X_i = Y_i \oplus pwd_i$ and $m_i^{'} = h(pwd_i\|X_i\|id_{ui})$.
2) Now $\mathbb{S}_c$ verifies $m_i^{'} = m_i$, if it holds, user information is considered valid, otherwise $\mathbb{S}_c$ terminates the session.
3) $\mathbb{U}_i$ chooses a new password pw_{ui}^{new} and scan new biometric B_i^{new}. subsequently, $\mathbb{S}_c$ computes $E_i^{new} = N_i \oplus h(id_{ui}\|pw_{ui}^{new})$, $g_i^{new} = Bi_{new} \oplus h(pw_{ui}^{new}\|id_{ui})$, $pwd_i^{new} = h(id_{ui}\|pw_{ui}^{new}\|N_i\|B_i^{new})$, $Y_i^{new} = Y_i \oplus pwd_i^{new} \oplus pwd_i$, $C_i^{new} = C_i \oplus pwd_i^{new} \oplus pwd_i$ and $m_i^{new} = h(X_i\|pwd_{ui}^{new}\|id_{ui})$. Finally, $\mathbb{S}_c$ updates the new computed values with previous values.

IV. ADVERSARIAL MODEL

An identical adversarial model as mentioned in [9], [29]–[32] has been adapted for the proposed protocol in this article with the following assumption:

1) The adversary $\mathbb{A}$ has unrestricted access to the public communication channel where $\mathbb{A}$ can replay and modify any message(s) as well as introduce a new message and can discard any message.
2) $\mathbb{A}$ can get the patient's private information like password or can steal his/her $\mathbb{S}_c$ but not both at the same time.
3) $\mathbb{A}$ can retrieve the $\mathbb{S}_c$'s stored parameters as mentioned [9], [32].

V. CRYPTANALYSIS OF THE BASELINE PROTOCOL

This section illustrates the weaknesses of baseline protocol. It has been shown that the baseline protocol is unable to provide user anonymity and also vulnerable to user Impersonation attack. For this purpose, an adversary $\mathbb{A}$ may act as a legal user and performs the steps as follows.

A. USER ANONYMITY VIOLATION ATTACK

This subsection analyzes the baseline protocol [28] and shows that it has been unable to provide user anonymity.

In the baseline protocol, the real identity of a user may be retrieved by another legitimate user of the system. It can be done by intercepting the Login request message that is normally transmitted over the insecure public communication channel.

Here a working example has been presented to clarify the logic behind the attack. Assume a legal user $\mathbb{U}_j$ wants to find the identity of another user $\mathbb{U}_i$. To retrieve the real identity of the user $\mathbb{U}_i$, $\mathbb{U}_j$ may perform the following steps.

1: $\mathbb{U}_j$ can extract the $\mathbb{S}_c$ values $\{Y_j, g_j, E_j, C_j, M_j, h(.)\}$ from his/her own $\mathbb{S}_c$ by using the methods mentioned in [9], [32].

2: Now $\mathbb{U}_j$ by using his/her own id_{uj}, pw_{uj}, B_j and N_j computes $pwd_j = h(id_{uj}\|pw_{uj}\|B_j\|N_j)$ and obtain the value $h(b\|x_s) = C_j \oplus pwd_j$.

3: When $\mathbb{U}_i$ sends a Login request $\{NID_i, H_1, G_i, t_1\}$ to the server $\mathbb{S}$ over the public communication channel, $\mathbb{U}_j$ intercepts the Login request as an eavesdropper.

4: Now by using the stolen smart card parameters he/she can compute $pwd_i = C_i \oplus h(b\|x_s)$, $D = pwd_i \oplus C_i$. Finally, $\mathbb{U}_j$ obtains the real identity id_{ui} of user's $\mathbb{U}_i$ as $id_{ui} = NID_i \oplus h(D)$.

B. USER IMPERSONATION ATTACK

For impersonation of a legitimate user, an attacker $\mathbb{A}$ retrieves the parameters stored in stolen $\mathbb{S}_c$ through power analysis mentioned in [9], [32]. Then, an adversary using $\mathbb{S}_c$ can easily masquerade legal user $\mathbb{U}_i$. The steps involved in the process are as follows:

1: $\mathbb{A}$ retrieves the concealed parameters stored in stolen $\mathbb{S}_c$. $\mathbb{A}$ also extract the real identity of a user as performed in section V-A. Now $\mathbb{A}$ by using his/her own identity id_{uj}, password pw_{uj}, biometric B_j and N_j can compute:

$$pwd_j = h(id_{uj}\|pw_{uj}\|B_j\|N_j) \tag{21}$$
$$h(b\|x_s) = C_j \oplus pwd_j \tag{22}$$

Now using C_i from the stolen smart-card, $\mathbb{A}$ can compute:

$$pwd_i = C_i \oplus h(b\|x_s) \tag{23}$$
$$X_i = Y_i \oplus pwd_i \tag{24}$$
$$Generate\ r_{u1} \tag{25}$$
$$D = pwd_i \oplus C_i \tag{26}$$
$$NID_i = id_{ui} \oplus h(D) \tag{27}$$
$$G_i = r_{u1} \oplus h(X_i\|id_{ui}\|D) \tag{28}$$
$$H_1 = h(id_{ui}\|X_i\|r_{u1}\|t_1) \tag{29}$$

$\mathbb{A}$ transmits request message $\{NID_i, H_1, G_i, t_1\}$.

2: Upon getting the message $\mathbb{S}$ computes:

$$t_2 - t_1 > \Delta t \tag{30}$$
$$id_{ui}^{'} = NID_i \oplus h(D) \tag{31}$$
$$X_i^{'} = h(id_{ui}^{'}\|x_s) \tag{32}$$
$$r_{u1}^{P\,'} = G_i \oplus h(id_{ui}^{'}\|X_i\|D) \tag{33}$$
$$H_1^{'} \stackrel{?}{=} h(id_{ui}^{'}\|D\|X^{'}\|r_{u1}^{'}\|t_1) \tag{34}$$
$$Generate\ r_{s1} \in Z_n^* \tag{35}$$
$$m_2 = r_{s1} \oplus h(id_{ui}^{'}\|X_i^{'}\|D) \tag{36}$$
$$H_2^{'} \stackrel{?}{=} h(r_{s1}\|X_i^{'}\|D\|id_{ui}^{'}\|t_3) \tag{37}$$

Now $\mathbb{S}$ conveys the message $\{m_2, H_2, t_3\}$ to $\mathbb{U}$

3: $\mathbb{A}$ computes by intercepting the message

$$r_{s1}^{'} = m_2 \oplus h(id_{ui}\|X_i\|D) \tag{38}$$
$$H_2^{'} = h(r_{s1}^{'}\|X_i\|D\|id_{ui}\|t_3) = H_2 \tag{39}$$
$$sk = h(X_i\|D\|id_{ui}\|r_{s1}^{'}\|r_{u1}) \tag{40}$$
$$H_3 = h(sk) \tag{41}$$

$\mathbb{A}$ sends his own message $\{H_3)\}$ to $\mathbb{S}$

4: At the arrival of the message, the server $\mathbb{S}$ may calculate a session key as $sk = h(X_i\|D\|id_{ui}^{'}\|r_{s1}\|r_{u1}^{'})$ and verify the message $H_3^{'} = h(sk)$. Hence, $\mathbb{A}$ has successfully impersonate the user $\mathbb{U}_i$ and already calculated a shared session key considered as legitimate key by $\mathbb{S}$.

VI. PROPOSED SCHEME

Based on the baseline protocol presented in [28], an enhance version has been introduced in this section consisting of three phases namely: registration, Login, and key agreement phases. Figure 2 shows the registration and key agreement phases of the proposed protocol.

A. REGISTRATION PHASE

1) To get registered, the user $\mathbb{U}_i$ along with a random number N_i also selects his/her identity id_{ui}, and password pw_{ui}. Now, computes $pwd_i = h(id_{ui}\|pw_{ui}\|N_i\|B_i)$ and sends $\{id_{ui}, pwd_i\}$ message using a secure channel to the server $\mathbb{S}$.

2) At the arrival of the message, the server $\mathbb{S}$ generates a random number $r_s \in Z_n^*$ and computes $X_i = h(id_{ui}\|x_s)$, $Y_i = X_i \oplus pwd_i$, $m_i = h(pwd_i\|X_i\|id_{ui})$ and $C_i = E_{x_s}(id_{ui}\|r_s) \oplus pwd_i$ and uses smartcard to store the calculated values. Finally, uses a secure network to send the smartcard $\mathbb{S}_c$ to the user.

3) Once the samrtcard arrives, the user $\mathbb{U}_i$ also computes $g_i = B_i \oplus h(id_{ui} \oplus pw_{ui})$, $E_i = N_i \oplus h(id_{ui} \oplus pw_{ui})$ and update these values into the $\mathbb{S}_c$. Now $\mathbb{S}_c$ contains $\{Y_i, g_i, E_i, C_i, m_i, h(.)\}$.

B. LOGIN PHASE

1: In the first step the user $\mathbb{U}_i$ puts his/her smartcard $\mathbb{S}_c$ in a card reader and enters the credentials including id_{ui}, pw_{ui} and imprints B_i using biometric reader. Now $\mathbb{S}_c$ computes $B_i = g_i \oplus h(id_{ui}\|pw_{ui})$ and checks if $d(B_i, B_i^*) \geq \tau$ holds then the session is terminated otherwise it calculates the following:

$$N_i = E_i \oplus h(id_{ui}\|pw_{ui}) \tag{42}$$
$$pwd_i = h(id_{ui}\|pw_{ui}\|N_i\|B_i) \tag{43}$$
$$X_i = Y_i \oplus pwd_i \tag{44}$$
$$m_i^{'} = h(pwd_i\|X_i\|id_{ui}) \tag{45}$$

$User\ \mathbb{U}_i$		$Server\ \mathbb{S}$
Registration		
Generate N_i		
Computes $pwd_i = h(id_{ui} \| pw_{ui} \| N_i \| B_i)$		
	$\xrightarrow{\{id_{ui}, pwd_i\}}$	
		Computes
		$X_i = h(id_{ui} \| x_s)$
		$Y_i = X_i \oplus pwd_i$
		$m_i = h(pwd_i \| X_i \| id_{ui})$
		Generates: $r_s \in Z_n^*$
		$\boxed{C_i = E_{x_s}(id_{ui} \| r_s) \oplus pwd_i}$
		$\mathbb{S}_c\ \{Y_i, C_i, h(.)\}$
	$\xleftarrow{\{Smart\ Card\}}$	
Compute		
$g_i = B_i \oplus h(id_{ui} \oplus pw_{ui})$		
$E_i = N_i \oplus h(id_{ui} \oplus pw_{ui})$		
$\mathbb{S}_c\ \{Y_i, G_i, E_i, C_i, M_i, h(.)\}$		

$User\ \mathbb{U}_i$		$Server\ \mathbb{S}$
Authentication		
Enters id_{ui}, pw_{ui}, B_i^*		
$B_i = g_i \oplus h(id_{ui} \| pw_{ui})$		
$d(B_i, B_i^*) \leq \tau$		
$N_i = E_i \oplus h(id_{ui} \| pw_{ui})$		
$pwd_i = h(id_{ui} \| pw_{ui} \| N_i \| B_i)$		
$X_i = Y_i \oplus pwd_i$		
$m_i^{'} = h(pwd_i \| X_i \| id_{ui})$ and checks $m_i^{'} \stackrel{?}{=} m_i$		
Generates $r_{u1} \in Z_n^*$		
$\boxed{NID_i = C_i \oplus pwdi}$		
$G_i = r_{u1} \oplus h(id_{ui} \| X_i)$		
$H_1 = h(id_{ui} \| X_i \| r_{u1} \| t_1)$		
	$\xrightarrow{\{NID_i, H_1, G_i, t_1\}}$	
		$t_2 - t_1 > \Delta t$
		$\boxed{(id_{ui} \| r_s) = D_{x_s}(NID_i)}$
		$X_i^{'} = h(id_{ui} \| x_s)$
		$\boxed{r_{u1} = G_i \oplus h(id_{ui} \| X_i)}$
		$\boxed{H_1^{'} \stackrel{?}{=} h(id_{ui} \| X^{'} \| t_1 \| r_{u1})}$
		Generate $r_{s1}, r_{s2} \in Z_n^*$
		$\boxed{m_2 = r_{s1} \oplus h(id_{ui} \| X_i)}$
		$\boxed{H_2^{'} \stackrel{?}{=} h(r_{s1} \| X_i \| id_{ui} \| t_3)}$
		$\boxed{NID_{new} = E_{x_s}(id_{ui} \| r_{s2}) \oplus r_{u1}}$
	$\xleftarrow{\{(m_2, H_2, t_3, NID_{new})\}}$	
$t_4 - t_3 > \Delta t$		
$\boxed{r_{s1}^{'} = m_2 \oplus h(id_{ui} \| X_i)}$		
$\boxed{H_2^{'} = h(r_{s1}^{'} \| X_i \| id_{ui} \| t_3) = H_2}$		
$\boxed{sk_{ui} = h(X_i \| id_{ui} \| r_{s1}^{'} \| r_{u1})}$		
$\boxed{C_i = NID_{new} \oplus r_{u1} \oplus pwd_i}$		
$H_3 = h(sk)$		
	$\xrightarrow{\{H_3\}}$	
		$\boxed{sk_s = h(X_i \| id_{ui}^{'} \| r_{s1} \| r_{u1})}$
		$H_3^{'} \stackrel{?}{=} h(sk) = H_3$
$\longleftarrow$	$\boxed{sk = sk_{ui} = sk_s}$	$\longrightarrow$

FIGURE 2. Proposed scheme.

Now verifies $m_i^{'} = m_i$ are equal then id_{ui} and pw_{ui} are considered valid values otherwise, $\mathbb{S}_c$ terminates the session. Furthermore, the smartcard $\mathbb{S}_c$ generates a random number $r_{u1} \in Z_n^*$ and calculates the following:

$$NID_i = C_i \oplus pwd_i \tag{46}$$
$$G_i = r_{u1} \oplus h(id_{ui} \| X_i) \tag{47}$$
$$H_1 = h(id_{ui} \| X_i \| r_{u1} \| t_1) \tag{48}$$

Now U_i sends request message $\{(NID_i, H_1, G_i, t_1)\}$ to $\mathbb{S}$.

C. KEY AGREEMENT PHASE

2: At the arrival of the request message, the server $\mathbb{S}$ first verifies its freshness using the time stamps $t_c - t_i = \Delta$ if it holds then computes the following:

$$(id_{ui} \| r_s) = D_{x_s}(NID_i) \tag{49}$$
$$X_i^{'} = h(id_{ui} \| x_s) \tag{50}$$
$$r_{u1}^{'} = G_i \oplus h(id_{ui} \| X_i) \tag{51}$$

Now verifies $H_1^{'} \stackrel{?}{=} h(id_{ui} \| X_i^{'} \| t_1 \| r_{u1}^{'})$ if it does not hold, the $\mathbb{S}$ terminates the session otherwise, it generates random numbers $r_{s1}, r_{s2} \in Z_n^*$ and computes:

$$m_2 = r_{s1} \oplus h(id_{ui} \| X_i) \tag{52}$$
$$H_2^{'} \stackrel{?}{=} h(r_{s1} \| X_i \| id_{ui} \| t_3) \tag{53}$$
$$NID_{new} = E_{x_s}(id_{ui} \| r_{s2}) \oplus r_{u1} \tag{54}$$

Now $\mathbb{S}$ sends the message $\{(m_2, H_2, t_3, NID_{new})\}$ to $\mathbb{U}_i$.

3: On receiving the message $\mathbb{U}_i$ checks the freshness of time stamps t_3 as $t_4 - t_3 = \Delta t$ and calculates:

$$r_{s1}^{'} = m_2 \oplus h(id_{ui} \| X_i) \tag{55}$$
$$verifies\ H_2^{'} = h(r_{s1}^{'} \| X_i \| id_{ui} \| t_3) = H_2 \tag{56}$$
$$sk_{ui} = h(X_i \| id_{ui} \| r_{s1}^{'} \| r_{u1}) \tag{57}$$
$$C_i = NID_{new} \oplus r_{u1} \oplus pwd_i \tag{58}$$
$$H_3 = h(sk) \tag{59}$$

Now transmits the message $\{H_3)\}$ to $\mathbb{S}$.

4: On receipt the $\mathbb{S}$ compute the session as $sk_s = h(X_i \| id_{ui}^{'} \| r_{s1} \| r_{u1})$ and checks the receiving parameter $H_3^{'} \stackrel{?}{=} h(sk) = H_3$. In case of failure the session is terminated by U_i while in case of success(true value) a shared session is key is calculated as $sk = sk_{ui} = sk_s$. This key is assumed to be the legitimate session key established between the user $\mathbb{U}_i$ and the server $\mathbb{S}$.

VII. SECURITY AND PERFORMANCE ANALYSIS

The proposed protocol must be analyzed to asses whether it is provably secure. This section presents a detailed analysis of the security and the performance of the proposed protocol. Security of the proposed system has been analyzed formally and informally whereas performance analysis has been performed using computation cost in terms of operation the number of operation executed as well as communication cost in terms of the messages(bits) exchanged during a single transaction of the protocol. Furthermore, the computation and communication cost has been compared to the existing protocol for validation. The analysis reveals that the proposed protocol is resistant to all known attacks as well as those attacks discussed in related protocols.

A. INFORMAL SECURITY ANALYSIS

The proposed protocol has been informally analyzed in this subsection against various security attacks.

1) ANONYMITY AND PRIVACY

An authentication protocol must ensure anonymity and privacy of the information. Using the proposed protocol, even if an adversary $\mathbb{A}$ intercepts the request message $\{NID_i, H_1, G_i, t_1\}$, $\mathbb{A}$ is unable to derive the identity id_{ui} of a user. The user identity id_{ui} is encrypted with private key x_s of the server that only known to the server. Therefore, it is impossible for $\mathbb{A}$ to extract id_{ui} from a message intercepted over the public channel. Therefore, the proposed protocol is not only anonymous but also protects user privacy.

2) OFFLINE PASSWORD GUESSING ATTACK

For this attack to take place, the adversary $\mathbb{A}$ must have both $\{Y_i, G_i, E_i, C_i, M_i, h(.)\}$ – the parameters stored in smartcard, and the request message $\{NID_i, H_1, G_i, t_i\}$. Even if the adversary $\mathbb{A}$ gets both, may be able to get the $pwd_i = NID_i \oplus C_i$. However, the password pw_{ui} is still secured through a one-way hash function as in $pwd_i = h(id_{ui} \| pw_{ui} \| N_i \| B_i)$ that makes it impossible for the attacker to revert it and compute the password. Therefore, the proposed scheme withstands password guessing attack.

3) RESIST REPLAY ATTACK

Any authentication protocol must protect against replay attack. In case of the proposed protocol, if an eavesdropper gets a hold of the request message $\{NID_i, H_1, G_i, t_1\}$ and tries to replay it. At the arrival of each message, the server $\mathbb{S}$ checks its freshness using the time stamp $t_c - t_i = \Delta$. If the time stamp is obsoleted, then $\mathcal{S}$ realizes that the message has been replayed and simply discard it. Furthermore, if $\mathbb{A}$ is able to generate a new time stamp t_a and use it to replay the request message. Then the adversary $\mathbb{A}$ also needs the identity id_{ui}, X_i, r_{u1} as well as t_1 to successfully compute the H_1. Due to the security provided by hash function, it is not possible for the adversary to extract those parameters. Even if $\mathbb{A}$ strives to replay the response message $\{m_2, H_2, NID_{new}, t_2\}$ from $\mathbb{S}$. Again in this case $\mathbb{U}_i$ first checks the freshness of the time stamp as well as needs r_{s1} to compute the correct value of H_2. However, deriving r_{u1} from H_1 is impossible due to the security of the hash function and only known to $\mathbb{U}_i$. Therefore, $\mathbb{A}$ fails to compute H_2. Hence, the proposed scheme resists against replay attack.

4) USER IMPERSONATION ATTACK

For launching impersonation attack, let suppose the adversary $\mathbb{A}$ intercepts the request message $\{NID_i, H_1, G_i, t_1\}$ and tries to impersonate $\mathbb{U}_i$. In this case the attacker must have access to the server secret key x_s in order to obtain $(id_{ui}\|r_s)$ by decrypting NID_i, however, x_s is only known to the server $\mathbb{S}$. Also, the attacker requires id_{ui}, X_i, r_{u1} and a valid time stamp t_1 to compute the H_1. Therefore, the proposed protocol is resilient to user impersonation attack.

5) SERVER IMPERSONATION ATTACK

Similarly, if the adversary $\mathbb{A}$ intercepts the response message and tries to send a fabricated message to $\mathbb{U}_i$ to impersonate the server. In this case the validity of the message is checked by the user $\mathbb{U}_i$ using its time stamp. Suppose, $\mathbb{A}$ successfully computes the r_{s1} using the smart-card stored parameters. However, without a valid time stamp, the attacker fails to compute the H_2. Eventually, $\mathbb{U}_i$ identifies $\mathbb{A}$ by verifying $H_2^{'} \stackrel{?}{=} H_2$. Consequently, the proposed protocol resilient to the server impersonation attack.

6) MUTUAL AUTHENTICATION

In the proposed protocol, in Login and authentication phases, first, the server $\mathbb{S}$ authenticates the $\mathbb{U}_i$ on receipt of the request message $\{NID_i, H_1, G_i, t_1\}$ from the $\mathbb{U}_i$. The authentication has been done by verifying the t_1 time stamp and comparing the $H_1^{'} = h(id_{ui}\|X_i\|r_{u1}\|t_1) \stackrel{?}{=} H_2$. Similarly, $\mathbb{S}$ sends back a response message $\{m_2, H_2, t_3, NID_{new}\}$ to $\mathbb{U}_i$. Here, $\mathbb{U}_i$ also authenticates the server $\mathbb{S}$ by verifying the freshness of the time stamp and validating $H_2^{'}$ by equating it with H_2. If both values are equal then the authentication is successful. Hence, mutual authentication is achieved by the proposed protocol.

7) PERFECT FORWARD SECRECY

The two random numbers r_{u1}, r_{s1} are involved in creation of the session key that are only known to $\mathbb{U}_i$ and $\mathbb{S}$, respectively. Suppose, id_{ui} and X_i gets compromised, still it is impossible for $\mathbb{A}$ to acquire both the random numbers r_{u1} and r_{s1}. Moreover, even if the attacker $\mathbb{A}$ is successful in compromising one session key, still it is impossible for the attacker to compute the new session key as both randoms numbers are chosen randomly by the $\mathbb{U}_i$ and $\mathbb{S}$ in each session. Therefore, the proposed scheme provides forward secrecy.

8) STOLEN VERIFIER ATTACKS

The proposed scheme creates or stores no verifier table in the server database. In case if there is a verifier table on the server database, an adversary having access to the server has the ability to retrieve the information from the verifier table and use it to impersonate a legitimate user. Hence, $\mathbb{A}$ tries to access and manipulate the server's verifier table [26]. On the other hand the proposed protocol stores no such verifier table on the server side, therefore, even an $\mathbb{A}$ with access to the server database is unable to obtain information of users' verifier.

9) DENIAL OF SERVICE

In the Login phase when $\mathbb{U}_i$ inputs his/her id_{ui}, pw_{ui} and scan B_i, before creating the Login message, the $\mathbb{S}_c$ verifies the validity of id_{ui},pw_{ui} and B_i. So, $\mathbb{S}_c$ computes $N_i = E_i \oplus h(id_{ui}\|pw_{ui}), pwd_i = h(id_{ui}\|pw_{ui}\|N_i\|B_i), X_i = Y_i \oplus pwd_i$ and $m_i^{'} = h(pwd_i\|X_i\|id_{ui})$. Then it checks verifies the two values of $m_1^{'} \stackrel{?}{=} m_1$. If the equation satisfies then the entered values are considered valid otherwise, $\mathbb{S}_c$ terminates the session. Hence, an adversary $\mathbb{A}$ is not capable of generating multiple Login requests in the network. Therefore, the proposed scheme withstands denial of service attack.

10) RESIST INSIDER ATTACK

The user $\mathbb{U}_i$ in the registration phase of the proposed protocol, sends $\{id_{ui}, pwd_i\}$ message to the server. The message contains the password that is protected by a one-way hash function as $pwd_i = h(id_{ui}\|pw_{ui}\|N_i\|B_i)$, where N_i is random number selected by $\mathbb{U}_i$. Since one-way hash functions are irreversible making it impossible for an insider to retrieve the password pw_{ui} of the user and random number N_i from this message. Therefore, the proposed protocol is protected against the insider attack.

11) SESSION KEY SECRECY

To set up a session key, the proposed protocol uses two random number r_{u1} and r_{s1} selected by the user $\mathbb{U}_i$ and server $\mathbb{S}$ respectively and independently for every session. So, the exposure of one session key does not make it possible for the attacker $\mathbb{A}$ to derive a new session key. Therefore, secrecy of the session is well protected in the by the proposed protocol.

12) MAN-IN-THE-MIDDLE ATTACK

The proposed scheme provides mutual authentication between user and server. The user is authenticated through parameter $H_1 = h(id_{ui}||X||t_1||r_{u1})$. The computation of H_1 is requires to compute the secret parameter $X_i = h(id_{ui}||x_s)$ of the user. The parameter X is stored in smart card by encrypting it with user password and biometrics. Therefore, to compute X_i, one requires the three factors including smart card, password and user biometrics. Hence, no adversary can compute X_i resulting non- computation of H_1 without three factors pertaining to user. Any adversary acting as man in middle can not compute H_1. Moreover, user authenticates the server using $H_2 = h(r_{s1}||X_i||id_{ui}||t_3)$. Similar to user part, the adversary again needs either X_i or sever secret key x_s to compute H_2. Therefore, no adversary can act as server. Furthermore, the computation of session key $sk_{ui} = h(X_i||id_{ui}||r_{s1}||r_{u1})$, which requires knowledge of user secret parameter X_i as well as the random number r_{s1}, r_{u1} generated by each participat i.e server and user. These random numbers cannot be exposed to any adversary acting as man in middle. Therefore, the proposed scheme strongly resists man in middle attack.

B. FORMAL SECURITY ANALYSIS

In this subsection, the security of the proposed protocol has been informally analyzed against all known attacks. It can be observed from the informal security analysis that the proposed protocol is protected against the known attacks. The analysis has been performed using a standard random oracle model (ROM). The formal analysis using ROM shows that the proposed protocol is provably secure. For this formal proof however, this article adopts similar model as presented in [9], [32]–[34].

Proof 1: Following are the oracles used in the formal security analysis of the proposed protocol:

- ***Reveal 1:*** This oracle will unconditional outcome of an input x from a one-way secure hash function $Y = h(x)$.
- ***Reveal 2:*** This oracle will unconditional results the plain text p from cipher text $C = E_k(p)$ without the knowledge of shared symmetric key k.

Theorem 1: Assuming that the h(.) one-way secure hash function and the symmetric encryption act as oracles. Then the proposed protocol *REBAKAS* is provably secure against an Adversary $\mathbb{A}$, to obtained the identity id_{ui} and the password pw_{ui} of the user $\mathbb{U}_i$ as well as the private key x_s of the server, and sk the session key shared between the user $\mathbb{U}_i$ and the server $\mathbb{S}$.

An imaginary adversary $\mathbb{A}$ has been created to perform experiment EXP1 using the two oracles *Reveal 1* and *Reveal 2* against the proposed protocol. The probability of success of the arithmetic algorithm is defined as:

$$Succ_1 = [Prbo[EXP1_{\mathcal{A},REBAKAS}^{HASH,ECDLP,SYMENC} = 1] - 1]. \quad (60)$$

Advantage of $\mathbb{A}$ performed series of queries q_{re1} and q_{re2} in polynomial time t with Adv1 as the success ratio as shown below

$$Adv1_{\mathbb{A},ASSAS}^{Hash,Ecdlp}(t_{exc}, q_{R1}, q_{R2}) = max_{\mathbb{A}}(succ_1) \quad (61)$$

If $\mathbb{A}$ successfully cracks the secure one-way hash function h(.) and get the plain text without the knowledge of shared symmetric key from $C = E_k(p)$. However, inverting a one-way has function is computationally infeasible and to derive the plain text from a symmetric operation without having a key. Hence, the proposed protocol is protected against the attacker $\mathbb{A}$ to derive/extract the id_{ui}, pw_{ui}, x_s, and sk.

Theorem 2: Each user $\mathbb{U}_i$ employes password from dictionary space of length $|L|$. Suppose l_h refers to the outcome length of hash, whereas, P_r refers to the introduced protocol for observing authentication. The Adversary $\mathbb{A}$ can launch various queries in polynomial time t. These queries include: *Send* query as q_s, *Execute* query as q_e and hash query as q_h. The advantage of $\mathbb{A}$ as *PA* can be substantiated as follows:

$$Adv2_P^{PA}(A) \leq \frac{q_h}{2^{l_h}} + \frac{(q_s+q_e)^2}{2(p-1)} + 2q_e.Adv2_A^{ECCDH}(H_1) + 2max\left\{\frac{q_h}{2^{l_h}}, \frac{q_s}{|L|}\right\} \quad (62)$$

Proof 2: The proof is elaborated with a flurry of games such as G_1 to G_3. The necessary assumptions are as follows: An event Suc_i refers to the correct guess Ω of $\mathbb{A}$ during G_i effectively in *Test*. With respect to the demand of our model, $\mathbb{A}$ is not supposed to determine the identity of user due to assumption of single user. The games for specifying proof are delineated as follows:

Game G_1: Within random oracle model it is observed as real protocol. Where, we choose randomly flipped coin value as Ω'. We have realized that the advantage of $\mathbb{A}$ to successfully predict Ω is as under:

$$Adv3_P^{PA}(A) = 2P_{rb}[Suc_0] - 1 \quad (63)$$

Game G_2: All oracles are executed against respective queries. Then a list is also maintained to observe the record (Rec, r) after executing the query given in security model. The hash query checks the record (Rec, r) to find any list, if it is found r is returned otherwise r' as random value is returned to the $\mathbb{A}$. From $\mathbb{A}$'s perspective G_1 and G_2 are indistinguishable through simulation, therefore,

$$P_{rb}[Suc_1] = P_{rb}[Suc_0] \quad (64)$$

Game G_3: During $G-3$, few collisions are avoided, which is terminated when few collision occur over values (G_i, m_2) along-with hash outcome. As $r_{u1}, r_{s1} \in [1, p-1]$, where the length of each hash value is l_h. Keeping in view the birthday paradox, then maximum collision probability for respective hash oracles is $\frac{q_h^2}{2^{l_h+1}}$. Whereas, the maximum collision probability for the value is $\frac{(q_s+q_e)^2}{2(p-1)}$. Therefore, we have:

$$P_{rb}[Suc_2] - P_{rb}[Suc_1] \leq \frac{q_h^2}{2^{l_h+1}} + \frac{(q_s+q_e)^2}{2(p-1)} \quad (65)$$

C. SECURITY ANALYSIS WITH PROVERIF

Security protocol are passed through verification to check if the protocol is immune to malicious attackers. Various tools are being used for verification of security protocols like AVISP, Skyther, and ProVerif etc. One tool that has been widely accepted for verification of security protocols is ProVerif [35]. It is used to check the resistance of a protocol against attacks, protection of privacy, and session key leakage. ProVerif is based on the famous *pi* calculus having ability to support various cryptographic operations.

The proposed protocol has also been formally analyzed with ProVerif tool to check its immunity against attacks, privacy and secret key leakage. The ProVerif code is with three parts are shown in Figure 3(a),(b) and (c), where Figure 3(a) specifies the declarations including constants, variables and the constructors, whereas Figure 3(b) codes the user and server processes part. Figure 3(c) shows implements the events and queries to verify correctness and secrecy of the session key. The results are shown in Figure 3(d). The simulation Result 1 and 2 show the proper initiation and termination of User and Server Processes, which verifies the reachability property of the proposed scheme; whereas, Results 3 shows

Algorithm 1 $EXP1^{HASH,ECDLP,SYMENC}_{\mathcal{A},RABACTMIS}$

Eavesdrop the request message $\{NID_i, H_1, G_i, t_1\}$, Where $H_1 = h(id_{ui}\|X_i\|r_{u1}\|t_1)$,$NID_i = Dx(NID_i)$,$G_i = r_{u1} \oplus h(X_i\|id_{ui})$
Call Reveal 1 on H_1 to obtain $id_{ui}^{'}, X_i^{'}, r_{u1}^{'}, t_1^{'} \leftarrow Reveal\ 1(H_1)$.
Call Reveal 1 on X_i to obtain $id_{ui}^{''}, x_s^{'} \leftarrow Reveal\ 1h(id_{ui}\|x_s)$
Call Reveal 2 on $D_{x_s}(NID_i)$ to obtain $id_{ui}^{'''}\|r_s^{'} \leftarrow\ Reveal\ 2\ NID_i$
Compute $r_{u1}^{''} = G_i \oplus h(id_{ui}^{'}\|X_i^{'})$.
if $(r_{u1})^{''} = (r_{u1})^{'}$ **then**
    Accept $id_{ui}^{'}$ as true identity of the user.
    Eavesdrop response message $\{m_2, H_2, t_3, NID_{new}\}$ where $m_2 = r_{s1} \oplus h(id_{ui}\|X_i)$, $H_2 = h(r_{s1}\|X_i\|id_{ui}\|t_3)$,$NID_{new} = E_{x_s}(id_{ui}\|r_s) \oplus r_{u1}$
    Compute $r_{s1} = m_2 \oplus h(id_{ui}^{'}\|X_i^{'})$
    Compute $sk^{'} = h(X_i^{'}\|id_{ui}^{'}\|r_{s1}^{'}\|r_{u1}^{'})$
    Call Reveal 2 on $E_{x_s}(NID_{new})$ to obtain $(id_{ui}^{''''}\|r_s)^{''} \leftarrow\ Reveal\ 2\ NID_{new}$
    if $r_s^{'} = r_s^{''}$ *then*
        Accept $x_s^{'}$ as private key of $\mathbb{S}$ and sk as correct shared session key between $\mathbb{U}_i$ and $\mathbb{S}$
    else
        return Fail
    end if
else
    return Fail
end if

TABLE 2. Comparison of the proposed protocol on the basis of computational cost.

Schemes →	Proposed	Omid et al. [28]	Mishra et al. [36]	Yan et al. [27]	Tan et al. [23]
Registration	$4t_{oh} + 1t_{E_s}$	$5t_{oh}$	$4t_{oh} + 1t_{E_s}$	$3t_{oh}$	$4t_{oh}$
Authentication	$15t_{oh} + 2t_{E_s}$	$16t_{oh}$	$10t_{oh} + 1t_{E_s}$	$11t_{oh}$	$12t_{oh} + 2t_{E_s}$
Total	$19t_{oh} + 3t_{E_s}$	$21t_{oh}$	$14t_{oh} + 2t_{E_s}$	$14t_{oh}$	$16t_{oh} + 2t_{E_s}$
Running Time	$\approx 0.0575ms$	$\approx 0.046ms$	$\approx 0.0414ms$	$\approx 0.0322ms$	$\approx 0.046ms$

that session key secrecy is maintained. Therefore, Proposed scheme is secure under ProVerif attack model.

D. PERFORMANCE ANALYSIS

Now that the security of the proposed protocol has been established formally and informally, it is time to analyze the protocol for performance in computation and communication. Here the proposed protocol has not only been analyzed for performance considering computation and communication overhead, but also been compared with existing state-of-the-art protocols including [23], [27], [28], [36]. Computation cost of the proposed and existing protocols has been computed using the number of operations times their frequency executed in one transaction of the protocol, whereas communication cost has been computed using the number of bit exchanged during one transaction of the protocol. For the sake of performance evaluation, only the registration and authentication phases are discussed. Following are the notation used to represent the cryptographic operations:

- t_{oh} : Computation time of a secure one-way hash function
- t_{E_s} : Computation time of a symmetric encryption/decryption

Omid et al.'s scheme takes $5t_{oh}$ operations during registration and $16t_{oh}$ during authentication processes, the scheme of Mishra et al. takes $4t_{oh} + 1t_{E_s}$ operations during registration and $10t_{oh} + 1t_{E_s}$ during authentication processes, Yan et al.'s scheme uses $3t_{oh}$ and $11t_{oh}$; whereas, Yan et al performs $4t_{oh}$ and $16t_{oh} + 2t_{E_s}$ for registration and authentication processes respectively.

Recently, Kilinc and Yanik [37] in their survey paper mentioned that the running time for executing a hash function t_{oh} is approximately 0.0023 *ms* and symmetric encryption/decryption t_{E_s} is approximately 0.0046 *ms*. Furthermore, according to Kilinc and Yanik, XOR and inverse operation are negligible in performance computation due to their insignificant execution time. The performance comparison has been shown in the Table 2 of the proposed protocol with recent related protocol presented in [23], [27], [28], [36].

This comparison has been pictorially depicted in FIGURE 4. It can be clearly seen from the Tables and Figures both that the proposed protocol has about 25% additional overhead in comparison to the Omid et al. protocol in case of computation cost, however, it is more secure as compared to rest of the schemes.

Similarly, Table 3 shows a comparison on the basis of communication cost using the number of messages exchanged in Login and authentication phases. Before proceeding ahead, it should be noted that the output of a one-way hash function and the length of a random number are 160 bits each and the

FIGURE 3. ProVerif simulation code.

TABLE 3. Comparison of the proposed protocol on the basis of communication cost.

Schemes	Comm. overhead(Bits)	Exchanged Messages
Proposed Scheme	1184	3
Omid et al. [28]	1024	3
Mishra et al. [36]	960	3
Yan et al. [27]	960	3
Tan et al. [23]	842	3

length of the time stamp is 32 bits, whereas the lenght of the user identity is also 160 bits.

From Table 3 it can be observed that in the proposed protocol, three messages are exchanged where the Login message $\{NID_i, H_1, G_i, t_1\}$ is equivalent to $(160 + 160 + 160 + 32) = 512 - bits$ whereas the authentication message $\{m_2, H_2, NID_{new}, t_3\}$ and $\{H_3\}$ is equivalent $(160 + 160 + 160 + 32 + 160) = 672 - bits$. So in total, the proposed scheme exchanges 1184 bits. The communication cost of related schemes has been shown in Table 3. The communication cost of the proposed protocol in comparison to existing protocols has been depicted in FIGURE. 5. During authentication process, Omid et al.'s schemes sends 1024 bits, Mishra et al's.and Yan et al.'s schemes transmit 960 bits, Tan et al.'s scheme sends 842 bits; whereas, the proposed communicates 1184 bits between user and sever. It can be observed that the proposed protocol bears 15.62% additional communication overhead in comparison to the base protocol of Omid et al. In terms of the number of the messages the proposed protocol has the same communication overhead as

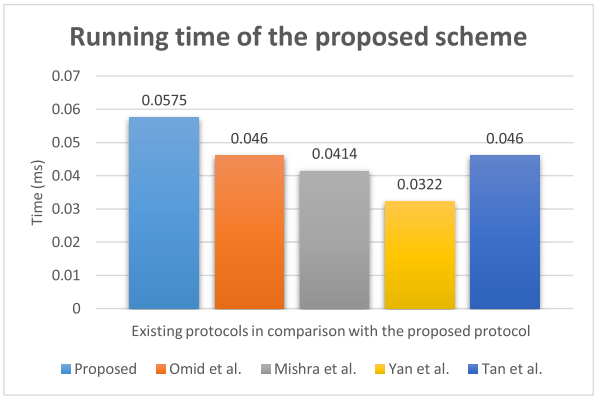

FIGURE 4. Computation cost comparison with other protocols.

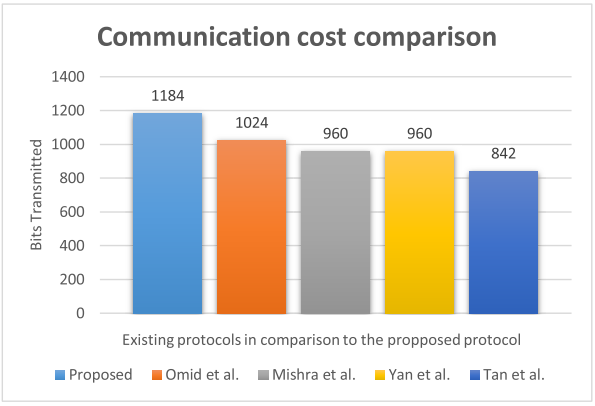

FIGURE 5. Communication cost comparison with other protocols.

TABLE 4. Comparison of the proposed protocol on the basis of security parameters.

Scheme: →	Proposed	[28]	[36]	[27]	[23]
Anonymity and Privacy	✓	✗	✗	✗	✗
Resist Replay Attack	✓	✓	✓	✓	✓
Stolen-Smart Card Attack	✓	✗	✓	✓	✗
Offline Password Guessing Attack	✓	✓	✓	✗	✗
Mutual Authentication	✓	✓	✓	✓	✓
Resist Insider Attack	✓	✗	✓	✓	✓
User Impersonation Attack	✓	✗	✓	✓	✓
Server Impersonation Attack	✓	✓	✓	✓	✓
Session Key Secrecy	✓	✓	✗	✓	✓
Perfect Forward Secrecy	✓	✓	✗	✗	✗
Denial of services	✓	✓	✓	✓	✗
Stolen Verifier Attack	✓	✓	✓	✓	✗
Man-in-the-Middle Attack	✓	N/A	N/A	N/A	N/A

the existing protocols. However, in terms of the number of bits exchanged during one transaction of the protocol, the proposed protocol has some additional overhead. The additional overhead is acceptable as it is leveraged to improve the security.

The proposed protocol has been comparatively analyzed using different security parameters. The comparison is presented in Table 4 where the table demonstrates the summarized security parameter comparison of the proposed protocol with the protocols presented in [23], [27], [28], [36]. Results show that the proposed protocol provides all the security features whereas its counterparts lacks some of the features. For example, compared to other related schemes the proposed protocol provides user anonymity and also resists user impersonation attack. It was observed to be successful in providing all the security features whereas none of its counterparts has been able to do so.

VIII. CONCLUSION

In this study, an investigation of security lapses in Omid et al.'s scheme has been performed and found that their protocol is exposed to an impersonation attack and also unable to protect user identity. Therefore, a robust and efficient scheme has been proposed to counter the issues of Omid et al.'s scheme. It has been shown through formal and informal analysis that the proposed protocol is provably secure against all possible attacks, including user impersonation and user anonymity attacks. The proposed protocol has also been compared with related state-of-the-art protocols on the basis of security requirements, computational and communication complexity where the proposed protocol presents superior results in terms of security and robustness. Hence, the proposed protocol is appropriate for TMIS.

REFERENCES

[1] 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.

[2] S. A. Chaudhry, H. Naqvi, and M. K. Khan, "An enhanced lightweight anonymous biometric based authentication scheme for TMIS," *Multimedia Tools Appl.*, vol. 77, no. 5, pp. 5503–5524, 2018.

[3] 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.

[4] S. A. Chaudhry, H. Naqvi, M. S. Farash, T. Shon, and M. Sher, "An improved and robust biometrics-based three factor authentication scheme for multiserver environments," *J. Supercomput.*, vol. 74, no. 8, pp. 3504–3520, Aug. 2018.

[5] M. Mohammedi, M. Omar, W. Aitabdelmalek, A. Mansouri, and A. Bouabdallah, "Secure and lightweight biometric-based remote patient authentication scheme for home healthcare systems," in *Proc. Int. Symp. Program. Syst. (ISPS)*, Apr. 2018, pp. 1–6.

[6] B. A. Alzahrani and A. Irshad, "A secure and efficient TMIS-based authentication scheme improved against Zhang et al.'s scheme," *Arabian J. Sci. Eng.*, vol. 43, no. 12, pp. 8239–8253, Dec. 2018.

[7] L. Zhang, Y. Zhang, H. Luo, and S. Tang, "Privacy protection for e-health systems by means of dynamic authentication and three-factor key agreement," *IEEE Trans. Ind. Electron.*, vol. 65, no. 3, pp. 2795–2805, Mar. 2018.

[8] M. Witteman, "Advances in smartcard security," *Inf. Secur. Bull.*, vol. 7, pp. 11–22, Jul. 2002.

[9] T. S. Messerges, E. A. Dabbish, and R. H. Sloan, "Examining smart-card security under the threat of power analysis attacks," *IEEE Trans. Comput.*, vol. 51, no. 5, pp. 541–552, May 2002.

[10] J.-Y. Liu, A.-M. Zhou, and M.-X. Gao, "A new mutual authentication scheme based on nonce and smart cards," *Comput. Commun.*, vol. 31, no. 10, pp. 2205–2209, 2008.

[11] T. F. Lee, J. B. Chang, C. W. Chan, and H. C. Liu, "Password-based mutual authentication scheme using smart cards," in *Proc. E-Learn. Inf. Technol. Symp. (EITS)*, Tainan, Taiwan, 2010.

[12] X. Xu, P. Zhu, Q. Wen, Z. Jin, H. Zhang, and L. He, "A secure and efficient authentication and key agreement scheme based on ECC for telecare medicine information systems," *J. Med. Syst.*, vol. 38, no. 1, pp. 1–7, 2014.

[13] Z.-Y. Wu, Y.-C. Lee, F. Lai, H.-C. Lee, and Y. Chung, "A secure authentication scheme for telecare medicine information systems," *J. Med. Syst.*, vol. 36, no. 3, pp. 1529–1535, Jun. 2012.

[14] D. He, C. Jianhua, and Z. Rui, "A more secure authentication scheme for telecare medicine information systems," *J. Med. Syst.*, vol. 36, no. 3, pp. 1989–1995, 2012.

[15] J. Wei, X. Hu, and W. Liu, "An improved authentication scheme for telecare medicine information systems," *J. Med. Syst.*, vol. 36, no. 6, pp. 3597–3604, 2012.

[16] Z. Zhu, "An efficient authentication scheme for telecare medicine information systems," *J. Med. Syst.*, vol. 36, no. 6, pp. 3833–3838, 2012.

[17] M. L. Das, A. Saxena, and V. P. Gulati, "A dynamic ID-based remote user authentication scheme," *IEEE Trans. Consum. Electron.*, vol. 50, no. 2, pp. 629–631, May 2004.

[18] M.-S. Hwang and L.-H. Li, "A new remote user authentication scheme using smart cards," *IEEE Trans. Consum. Electron.*, vol. 46, no. 1, pp. 28–30, Feb. 2000.

[19] M. Sandirigama, A. Shimizu, and M. T. Noda, "Simple and secure password authentication protocol (SAS)," *IEICE Trans. Commun.*, vol. 83, no. 6, pp. 1363–1365, 2000.

[20] H. Arshad and M. Nikooghadam, "An efficient and secure authentication and key agreement scheme for session initiation protocol using ECC," *Multimedia Tools Appl.*, vol. 75, no. 1, pp. 181–197, Jan. 2016.

[21] D. Guo, Q. Wen, W. Li, H. Zhang, and Z. Jin, "An improved biometrics-based authentication scheme for telecare medical information systems," *J. Med. Syst.*, vol. 39, p. 20, Mar. 2015.

[22] A. K. Awasthi and K. Srivastava, "A biometric authentication scheme for telecare medicine information systems with nonce," *J. Med. Syst.*, vol. 37, p. 9964, Oct. 2013.

[23] Z. Tan, "An efficient biometrics-based authentication scheme for telecare medicine information systems," *Network*, vol. 2, no. 3, pp. 200–204, 2013.

[24] C.-T. Li and M.-S. Hwang, "An efficient biometrics-based remote user authentication scheme using smart cards," *J. Netw. Comput. Appl.*, vol. 33, no. 1, pp. 1–5, Jan. 2010.

[25] R. Amin and G. P. Biswas, "A secure three-factor user authentication and key agreement protocol for tmis with user anonymity," *J. Med. Syst.*, vol. 39, p. 78, Aug. 2015.

[26] H. Arshad and M. Nikooghadam, "Three-factor anonymous authentication and key agreement scheme for telecare medicine information systems," *J. Med. Syst.*, vol. 38, p. 136, Dec. 2014.

[27] X. Yan, W. Li, P. Li, J. Wang, X. Hao, and P. Gong, "A secure biometrics-based authentication scheme for telecare medicine information systems," *J. Med. Syst.*, vol. 37, p. 9972, Oct. 2013.

[28] O. Mir and M. Nikooghadam, "A secure biometrics based authentication with key agreement scheme in telemedicine networks for e-health services," *Wireless Pers. Commun.*, vol. 83, no. 4, pp. 2439–2461, 2015.

[29] T. Eisenbarth, T. Kasper, A. Moradi, C. Paar, M. Salmasizadeh, and M. T. M. Shalmani, "On the power of power analysis in the real world: A complete break of the KeeLoq code hopping scheme," in *Advances in Cryptology—CRYPTO*. Berlin, Germany: Springer, 2008, pp. 203–220.

[30] W.-H. Yang and S.-P. Shieh, "Password authentication schemes with smart cards," *Comput. Secur.*, vol. 18, no. 8, pp. 727–733, 1999.

[31] M. Hölbl, T. Welzer, and B. Brumen, "An improved two-party identity-based authenticated key agreement protocol using pairings," *J. Comput. Syst. Sci.*, vol. 78, no. 1, pp. 142–150, Jan. 2012.

[32] P. Kocher, J. Jaffe, and B. Jun, "Differential power analysis," in *Advances in Cryptology—CRYPTO*. Berlin, Germany: Springer, 1999, pp. 388–397.

[33] V. Odelu, A. K. Das, and A. Goswami, "A secure effective key management scheme for dynamic access control in a large leaf class hierarchy," *Inf. Sci.*, vol. 269, pp. 270–285, Jun. 2014.

[34] S. Chatterjee, A. K. Das, and J. K. Sing, "An enhanced access control scheme in wireless sensor networks," *Adhoc Sensor Wireless Netw.*, vol. 21, no. 1, pp. 121–149, 2014.

[35] A. Irshad, M. Sher, B. A. Alzahrani, A. Albeshri, S. A. Chaudhry, and S. Kumari, "Cryptanalysis and improvement of a multi-server authentication protocol by lu," *KSII Trans. Internet Inf. Syst.*, vol. 12, no. 1, pp. 1–27, Jan. 2018.

[36] D. Mishra, S. Mukhopadhyay, A. Chaturvedi, S. Kumari, and M. K. Khan, "Cryptanalysis and improvement of Yan et al.'s biometric-based authentication scheme for telecare medicine information systems," *J. Med. Syst.*, vol. 38, p. 24, Jun. 2014.

[37] H. H. Kilinc and T. Yanik, "A survey of SIP authentication and key agreement schemes," *IEEE Commun. Surveys Tuts.*, vol. 16, no. 2, pp. 1005–1023, 2nd Quart., 2014.

ZAHID MEHMOOD received the Ph.D. degree in cyberspace security from Shanghai Jiao Tong University, Shanghai, in 2018, and the M.S. degree in computer science from the Department of Computer Science and Software Engineering, International Islamic University Islamabad, Pakistan, in 2010, where he is currently the Deputy Director of the Exam and Automation Department. He has also been involved in development and management of the Aljamia Student Information System, International Islamic University Islamabad. His broad research interests include elliptic curve cryptography, SIP authentication, security of the Internet of Things, and authentication protocols specifically based on lightweight cryptosystems.

ANWAR GHANI received the B.S. degree in computer science from the University of Malakand, Pakistan, in 2007, and the M.S. and Ph.D. degrees in computer science from the Department of Computer Science and Software Engineering, International Islamic University Islamabad, in 2016 and 2011, respectively, where he is currently a Faculty Member of the Department of Computer Science and Software Engineering. He was a Software Engineer with Bioman Technologies, from 2007 to 2011. He was selected as an Exchange Student through the EURECA Program with VU University Amsterdam, The Netherlands, in 2009, and the EXPERT Program with Masaryk University, Brno, Czech Republic, in 2011, supported by the EUROPEAN Commission. His broad research interests include information security, wireless sensor networks, next generation networks, and energy efficient collaborative communication.

GONGLIANG CHEN received the M.S. degree in mathematics from the Institute of Applied Mathematics, Academy of Sciences of China, in 1986, and the Ph.D. degree in mathematics from the University de Saint-Etienne, France, in 1993. He is currently a Professor/Doctoral supervisor with the School of Cyber Security (SCS), SJTU-ParisTech Elite Institute of Technology, Shanghai Jiao tong University (SJTU). He is a Visiting Fellow/Doctoral Supervisor with the Institute of Information Engineering, Chinese Academy of Sciences (IIECAS). His research interests include network security, information security, the security of RFID, lightweight cryptology, and cryptography technology and its applications.

AHMED S. ALGHAMDI graduated from the Department of Electrical Engineering and Computer Science, School of Engineering, The Catholic University of America, Washington, DC, USA, with an emphasis on network security, and the M.Sc. degree in information systems from DePaul University, Chicago, in 2010. He is currently an Assistant Professor with the Department of Cybersecurity, University of Jeddah.

Prior to entering academia, he has practiced IT for over six years. He started the technical career as a Platform Specialist and eventually became a Network Administrator with Batelco Jeraisy Ltd., Atheer, Inc., in 2003. He then practiced teaching with the Royal Commission in Yanbu for two years before he became an IT Manager with Al Musahim Gate Company, in 2006.