Chaotic Map based Privacy Preservation User Authentication Scheme for WBANs

Shreeya Swagatika Sahoo
National Institute of Technology
Rourkela, India
Email: shreeya.swagatika@gmail.com

Sujata Mohanty
National Institute of Technology
Rourkela, India
Email: sujatam@nitrkl.ac.in

Abstract—With the advancement of network and communication technologies, the Internet plays a vital role in our day-to-day life. These technologies provide several online services such as online booking, gaming, online shopping, e-health care, etc. WBAN such type of e-healthcare systems which consists of several sensors implanted in patient's body. These sensor nodes collect various health-related sensitive data such as heart rate, blood pressure, pulse rate, etc. and send them to the network server for processing However, as the information is communicated through the open channel, security and privacy of the shared information remain a paramount concern. Thus, authentication is essential to verify the legitimacy of the communicating parties. In this work, we proposed a mutual authentication system for WBANs. The formal security analysis of the proposed scheme has been done using the Real-Or-Random Model (ROR). In addition, the informal security analysis of the scheme proves that it can withstand several known attacks. Moreover, the scheme is also efficient in terms of computation and communication cost compared to other existing schemes.

Index Terms—Index Terms: Mutual Authentication, Chebyshev chaotic map, WBAN, Real-or-Random.

I. Introduction

Due to the rapid development of wireless communication technologies, health care services have been remarkably promoted. WBAN such type of e-healthcare systems which consists of several sensors implanted in patient's body. These sensor nodes collect various health-related sensitive data such as heart rate, blood pressure, pulse rate, etc. and send them to the cloud server for processing. However, the leakage of sensitive data of the patient cause threats leading to tampering of health or may cause death. Thus, preserving the patient's privacy is a crucial feature in the health care system. For secure communication, both patient and application provider should authenticate each other before sending the data.

Lamport introduced the simple password-based authentication scheme for a single server environment (SSE) [1]. Afterwards, several smart card and biometric-based authentication schemes have been suggested for SSE [2]–[5]. However, in an SSE, the user needs to register with every server whereas in a multiserver environment (MSE) the user has to register once. Compared to a single server, MSE offers better and extensive services to the users. Later, many authentication schemes have been suggested for MSE based on ECC, RSA, Chebyshev chaotic map, etc. [6]–[10]. Chaotic map-based authentication scheme has been studied widely due to its better performance than traditional cryptography.

Xiao et al. designed an efficient chaotic based authentication scheme for deniable authentication [11]. Later, Han pointed out that the scheme in [11] suffers their new attack that is the session key is compromised even though an adversary cannot get any secret key [12]. Yoon et al. pointed out that Xiao et al.'s scheme is susceptible to off-line password guessing (OPG) attack and suggested a new scheme which can secure against Han et al.'s attack [13]. Guo et al. suggested a secure group key agreement authentication scheme based on the chaotic map [14]. Later, He cryptanalysis the Guo et al.'s scheme and pointed out that the scheme is vulnerable to an OPG attack [15].

Tsaur et al. suggested a self-verified timestamp technique to overcome the issues of clock synchronization problem [16]. Later, Lee et al. present an improved scheme pointing that Tsaur et al.'s scheme could not achieve insider attack and known-plaintext attack [17]. Also, the scheme could not achieve user anonymity (UA) and perfect forward secrecy. Later, Li et al. pointed out that Lee et al's scheme suffers from server spoofing attack, registration spoofing attack, inefficient detection of unauthorized login, and not supporting password change phase [18]. Then, they suggested an extended chaotic map and dynamic ID-based authentication scheme which can ensure UA and resist several well-known attacks. Nevertheless, the scheme still insecure and susceptible to smart card loss attack, insider attack, user impersonation attack, and session-specific temporary information attack [19]. In this paper, we have proposed an authentication scheme for WBAN to solve the weaknesses of the previous scheme. Further, the formal security analysis proves that the PUASW can resist several known attacks. The scheme is efficient in terms of computational and communicational cost.

The remainder of the work is sketched as follows: Next Section demonstrates the necessary mathematical preliminaries. A privacy preservation user authentication scheme is presented in Section III. In section IV, the formal security analysis of the PUASW is done using Real-or-

Random (ROR) model. The formal security verification and performance analysis of the PUASW are presented in Section V and Section VI respectively. Lastly, we conclude the paper in Section VII.

II. Preliminaries

The mathematical preliminaries such as hash function, Chebyshev chaotic map, and adversary model are presented in this section.

A. Hash function

A one-way cryptographic hash function $h : \{0,1\}^* \rightarrow \{0,1\}^n$, produces a fixed length output string of any arbitrary length input string. The probability of an adversary in finding collision is defined as $Adv_A^{hash}(t_1) = Prob[((\upsilon, \upsilon') , \upsilon \neq \upsilon') : h(\upsilon) = h(\upsilon')]$.

B. Chebyshev chaotic map

Definition 1: The Chebyshev polynomial $T_w(\rho) : [-1,1] \rightarrow [-1,1]$ of degree w is defined as $T_w(\rho) = cos(w.cos^{-1}(\rho))$ if $\rho \ \varepsilon \ [-1,1]$ and $cos(w\theta)$ if $\rho = cos\theta$, where $\theta \ \varepsilon \ [0,\pi]$. The recurrence relation of $T_w(\rho)$ is defined as $T_w(\rho) = (2\rho T_{(w-1)}(\rho) - T_{(w-2)}(\rho)) \bmod p$, where $w \geq 2$ and $T_0(\rho) = 1, T_1(\rho) = \rho$, and p is a high entropy prime number.

Definition 2: The semi-group property of chebyshev chaotic map is defined as $T_\Psi(T_\Phi(\rho)) = T_\Phi(T_\Psi(\rho))$, where $\rho \ \varepsilon \ [-\infty, +\infty]$ and Ψ, Φ are two random positive numbers.

Definition 3: Computational Diffie-Hellman problem is defined as the computation of $T_{\Psi\Phi}(\rho)$ is hard, although T_Ψ, T_Φ, and ρ is given. In DLP problem, if T_Ψ and ρ are given, it is hard to compute Ψ.

C. Adversary model

An adversary model has the following capabilities.

1) The communication over open channel leads to various passive and active attacks [20].
2) An adversary may insert, delete, or modify the communicated message.
3) Using power analysis attack, the smart card information can be extracted [21], [22].
4) An adversary can be an insider or outsider.

III. Proposed scheme

We suggest a privacy-preserving user authentication scheme for WBANs. The PUASW consists of three entities such as user/client, application provider, network manager. Here, we consider the network server as a trusted party. The notation used in this paper is provided in Table I.

TABLE I: Notations and terminology

Notation	Description
U_c	c^{th} User
NS	Network Server
AP_n	Application Provider
ID_c	User Identity
PW_c	User Password
AID_n	AP_n Identity
SID	NS Identity
A_v	Adversary
x, y	Master key of NS
SK	Session key
$\triangle T$	Maximum transmission delay

A. Registration phase

The registration of U_c and AP_n with network manager NS has been done using the following steps.

1) User registration phase: The U_c generates his identity ID_c, password PW_c, and a random number b. Then, compute $PW_{i1} = h(ID_c \| PW_c \| b)$ and sends $\{ID_c, PW_{i1}\}$ to the network server. NS computes $A_n = h(ID_c \| x \| R_1)$, $B_n = h(A_n \| PW_{i1})$, $C_n = A_n \oplus (ID_c \| PW_{i1})$ where x is the master key and R_1 is the random number. Then, NS sends $\{B_n, C_n\}$ to the U_c through open channel.

2) AP_n registration phase: AP_n submits its identity AID_n to the NS in a secure channel. Now, NS generates a master key y and computes $S_1 = h(AID_n \| y)$, $S_2 = y \oplus SID$ sends $\{S_1, S_2\}$ to the application provider through a secure channel. The details of the registration phase are described in Table II.

B. Login phase

The U_c perform the complying login activities to access the services from the AP_n.

1) U_c enters the ID_c, PW_c and computes $PW_{i1}^* = h(ID_c \| PW_c \| b)$, $A_n^* = C_n \oplus h(ID_c \| PW_{i1}^*)$. Then, checks $B_n^* \stackrel{?}{=} h(A_n^* \| PW_{i1}^*)$. If true, generate a random number n_1 and computes $L_1 = T_{n_1}(A_n) \bmod p$, $L_2 = h(ID_c \| A_n^* \| B_n^*)$, $UID_i = h(ID_c \| L_2 \| A_n^* \| T_u)$, $NID_i = E_x(ID_c, L_1)$. Now, the message $\{M_1 = \{UID_i, NID_i, L_2, T_u\}$ is send to the NS, where T_u is the time stamp.
2) NS checks the validity of the time stamp $|T_n - T_u| \leq \triangle T$. If it is valid, then decrypt NID_i as $D_x(NID_i) = (ID_c, L_1)$ and compute $L_2^* \stackrel{?}{=} h(ID_c \| A_n \| B_n)$. If true, then computes $CID_i = h(AID_n \| S_1 \| h(y))$, $RID_i = E_{(S_1 \| y)}(A_n, L_1)$. Then, SN sends the authentication message $M_2 = \{CID_i, RID_i, T_n\}$ to the AP_n through an insecure channel.

C. Authentication phase

AP_n receives the login request message from the NS and both U_c and AP_n mutually authenticated to each other. Graphical representation of the login and authentication phase (LAP) are shown in Table III. The steps are performed as follows.

1) AP_n verifies the time stamp $|T_a - T_n| \leq \triangle T$, after receiving the login message $\{M_2\}$ at time T_a. If

TABLE II: Registration phase of the PUASW

User(U_c)		Network Server(NS)
Choose ID_c, PW_c and random number b.		
$PW_{i1} = h(ID_c \| PW_c \| b)$		
	$\xrightarrow{\{ID_c, PW_{i1}\}}$	
		Generate random number R_1
		$A_n = h(ID_c \| x \| R_1)$
		$B_n = h(A_n \| PW_{i1})$
		$C_n = A_n \oplus h(ID_c \| PW_{i1})$
		Stores $\{A_n, B_n\}$
	$\xleftarrow{\{B_n, C_n, h(.)\}}$	
Stores $\{B_n, C_n, b\}$		
Application Provider(AP_n)		Network Server(NS)
Choose AID_n		
	$\xrightarrow{\{AID_n\}}$	
		Generate master key y
		$S_1 = h(AID_n \| y)$
		$S_2 = y \oplus SID$
	$\xleftarrow{\{S_1, S_2\}}$	
Stores $\{S_1, S_2\}$		
		Secure channel $\longrightarrow$

it holds, then AP_n calculates $y = S_2 \oplus SID$ and compare $CID_i^* \stackrel{?}{=} h(AID_n \| S_1 \| h(y))$. If the equality fails, AP_n abort the session. Otherwise, generate a random number n_3 and calculate $D_{(S_1 \| y)}(RID_i) = (A_n, L_1, T_{n_1})$, $R_1 = T_{n_3}(A_n) \bmod p$, $SK = T_{n_1}(R_1)$, $R_2 = h(SID \| A_n \| SK)$, $R_3 = E_{(A_n \| L_1)}(T_{n_3}, SID)$. Now, AP_n transmits authentication message $M_3 = \{R_2, R_3, T_n\}$ to the U_c.

2) The U_c verifies T_a and if it satisfies, then decrypt R_3 as $D_{(A_n \| L_1)}(R_3) = (T_{n_3}, SID)$. The U_c computes $SK^* = T_{n_3}(L_1)$, $R_2^* \stackrel{?}{=} h(SID \| A_n \| SK^*)$. If the condition true, AP_n is authenticated and U_c computes $ACK = h(ID_c \| SID \| SK^*)$, $R_4 = ID_c \oplus SK^*$. Then, the message $M_4 = \{ACK, R_4, T_u^*\}$ is send to the AP_n through an open channel.
3) To complete the mutual authentication, AP_n first checks the time stamp and then compute $ID_c^* = R_4 \oplus SK^*$, $ACK^* \stackrel{?}{=} h(ID_c^* \| SID \| SK)$. If the condition satisfied, then both are mutually authenticated otherwise AP_n abort the session. Now, both U_c and AP_n use their session key for communication.

D. Password change phase

A legal user U_c could change or update his current password to a new password as follows.

1) U_c inserts his SC and puts his identity ID_c and password PW_c. SC calculates $PW_{i1}^* = h(ID_c \| PW_c \| b)$, $A_n^* = C_n \oplus h(ID_c \| PW_{i1}^*)$, and checks $B_n^* \stackrel{?}{=} h(A_n^* \| PW_{i1}^*)$. If the comparison satisfies, then it ask for the new password PW_c^{new}. Otherwise, reject the session.
2) SC computes $C_n^{new} = A_n \oplus (ID_c \| PW_c^{new})$, $B_n^{new} = h(A_n \| PW_c^{new})$ and replaces C_m with C_m^{new} and B_m with B_m^{new}.

IV. Security analysis of the proposed scheme

The formal and informal security analysis of the PUASW describes as follows.

A. Formal security proof

The formal security of the PUASW is analyzed under Real-Or-Random (ROR) model [23]. This model is one of the standard models to prove session key security of the protocol. The definitions are depicted as follows.

Participants : We denote $\Pi_{u_m}^{\Psi}$, Π_{nm}^{γ}, Π_{ap}^{β} as the instance of Ψ, γ, β of entities user U_c, network manager NS, and application provider AP_n respectively, which are also known as oracle. Π^t is the union set of all participants where instance t is an oracle.

Partnering : Two instances Π^{t1} and Π^{t2} are to be partnered if both U_c and AP_n share the same session identifications (sid_i) and are in accepted state.

Adversary : An adversary A_v uses the Dolev-Yao model in which he has control over the communication. In the ROR model, to get the session key or forge the message A_v makes the queries to the simulator.

- $Execute(\Pi_{u_m}^{\Psi}, \Pi_{nm}^{\gamma}, \Pi_{ap}^{\beta})$: This query simulates the eavesdropping attack which permits A_v to read the communicated message among the participants in login and authentication phase.
- $Send(\Pi^t, m)$: This query models as an active attack in which A_v can listen, modify, or delete a message. With this oracle query, an instance Π^t will generate the output after receiving the message m.
- $Reveal(\Pi^t)$: This query discloses the session key generated by Π^t and its partner.
- $Test(\Pi_t)$: This query models the semantic security of SK between U_c and AP_n. An adversary flipping an unbiased coin f in $Test$ query. If the the flipping value $f = 0$, then returns an arbitrary number and if $f = 1$, returns session key for instance Π^t.

Semantic security: In the ROR model, an adversary is allowed to ask several $Test$ queries to different instances. All the $Test$ queries will be answered and returns the same value for guessed bit f. The A_v is considered successful if he guesses f correctly. Suppose, S denotes the event which A_v wins. The advantage of A_v to breaching the semantic security of the proposed authentication scheme ($PUASW$) is $Adv_p{}^{PUASW}(A_v) = |2.Prob[SE_0] - 1|$.

Theorem 1 : Let A_v denotes an adversary running in polynomial time t_m against the proposed scheme p. The queries q_h, q_s, q_e denote the number of hash queries, send queries, and execution queries respectively. $|2^l|$ represents the range space of the hash function and q represents a big prime order. q_t is the A_v's number of guess attempt towards the CS. The advantage of breaking the SK-security of the A_v is

$$Adv_p^{PUASW}(A_v) \leq \frac{q_h^2}{2^l} + \frac{(q_s + q_e)^2}{(q-1)} + \frac{2q_s}{2^l}$$

TABLE III: Login and authentication phase of the PUASW

User(U_c)	Network Server(NS)	Application Provider(AP_n)
U_c enters ID_c, PW_c SC Compute $PW^*_{i1} = h(PW_c \| b)$ $A^*_n = C_n \oplus h(ID_c \| PW^*_{i1})$ $B^*_n \stackrel{?}{=} h(A^*_n \| PW^*_{i1})$ If holds, generates n_1 and compute $L_1 = T_{n_1}(A_n) \bmod p$ $L_2 = h(ID_c \| A^*_n \| B^*_n)$ $UID_i = h(ID_c \| L_2 \| A^*_n \| T_u)$ $NID_i = E_x(ID_c, L_1)$		
$M_1 = \{UID_i, NID_i, L_2, T_u\} \longrightarrow$		
	Check $\|T_n - T_u\| \leq \triangle T$ Generate a random number v $D_x(NID_i) = (ID_c, L_1)$ $L^*_2 \stackrel{?}{=} h(ID_c \| A_n \| B_n)$ If true, then computes $CID_i = h(AID_n \| S_1 \| h(y))$ $RID_i = E_{(S_1 \| y)}(A_n, L_1)$	
	$M_2 = \{CID_i, RID_i, T_n\} \longrightarrow$	
		Check $\|T_a - T_n\| \leq \triangle T$ Compute $y = S_2 \oplus SID$ $CID^*_i \stackrel{?}{=} h(AID_n \| S_1 \| h(y))$ If true, then generate n_3 and calculate $D_{(S_1 \| y)}(RID_i) = (A_n, L_1, T_{n_1})$ $R_1 = T_{n_3}(A_n) \bmod p$, $SK = T_{n_1}(R_1)$ $R_2 = h(SID \| A_n \| SK)$ $R_3 = E_{(A_n \| L_1)}(T_{n_3}, SID)$
	$\longleftarrow M_3 = \{R_2, R_3, T_a\}$	
Check $\|T^*_u - T_a\| \leq \triangle T$ $D_{(A_n \| L_1)}(R_3) = (T_{n_3}, SID)$ $SK^* = T_{n_3}(L_1)$ $R^*_2 \stackrel{?}{=} h(SID \| A_n \| SK^*)$ If true, then computes $ACK = h(ID_c \| AID_n \| SK^*)$ $R_4 = ID_c \oplus SK^*$		
$M_5 = \{ACK, R_4, T^*_u\} \longrightarrow$		
		Check $\|T^*_a - T^*_u\| \leq \triangle T$ $ID^*_c = R_4 \oplus SK^*$ $ACK^* \stackrel{?}{=} h(ID^*_c \| SID \| SK)$
		Insecure channel $\longleftarrow$

Proof : We define a set of games $Game_i$, where $i = \{0, 1, 2, 3\}$. The game start with $Game_0$ which is the real attack and ends with $Game_3$. Let, SE_i be an event, where A_v guesses the bit f successfully in game G_i.

$Game_0$: This game is the actual attack against the PUASW in the random oracle model. By definition, we have

$$Adv_p^{PUASW}(A_v) = |2.Prob[SE_0] - 1| \tag{1}$$

$Game_1$: This game corresponds to an eavesdropping attack by querying the $Execute(U^\alpha_i, NS^\gamma, AP^\beta_n)$ oracle. Finally, A_v quired the $Test$ oracle and match the session key. The random number is used to compute the session key for which A_v could not break the authentication process. Thus, $Game_0$ and $Game_1$ are indistinguishable. So, we have

$$Prob[SE_0] = Prob[SE_1] \tag{2}$$

$Game_2$: This game is modeled as an active attack which considers all oracles simulated in the $Game_0$. There are two cases of a collision such as hash function and a random number. According to birthday paradox, the upper bound of the probability of collision in hash function and random number in different session are $\frac{q_h^2}{2^{l+1}}$ and $\frac{q_s^2 + q_e^2}{2(q-1)}$ respectively. Hence, we obtain

$$|Prob[SE_2] - Prob[SE_1]| \leq \frac{q_h^2}{2^{l+1}} + \frac{(q_s + q_e)^2}{2(q-1)} \tag{3}$$

$Game_3$: This game simulates forging of transmitted login and authentication messages. If the submitted messages was not previously queried, then an adversary loose the game. Thus, the probability of forge the message is $\frac{q_s}{2^l}$. Then, we obtain,

$$|Prob[SE_3] - Prob[SE_2]| \leq \frac{q_s}{2^l} \tag{4}$$

Finally, when all queries are made by an adversary, he has only choice to guess the bit c by $Test$ query in the final game, we have

$$Prob[SE_3] = 1/2 \tag{5}$$

From equation 3 and 4, we obtain

$$|Prob[SE_3] - Prob[SE_1]| \leq \frac{q_h^2}{2^{l+1}} + \frac{(q_s + q_e)^2}{2(q-1)} + \frac{q_s}{2^l} \tag{6}$$

From equation 2 and 6

$$|Prob[SE_3] - Prob[SE_0]| \leq \frac{q_h^2}{2^{l+1}} + \frac{(q_s+q_e)^2}{2(q-1)} + \frac{q_s}{2^l} \quad (7)$$

From equation 5 and 7

$$|2Prob[SE_0] - 1| \leq 2(\frac{q_h^2}{2^{l+1}} + \frac{(q_s+q_e)^2}{2(q-1)} + \frac{q_s}{2^l}) \quad (8)$$

$$|2Prob[SE_0] - 1| \leq \frac{q_h^2}{2^l} + \frac{(q_s+q_e)^2}{(q-1)} + \frac{2q_s}{2^l}) \quad (9)$$

From equation 1 and 9, we get

$$Adv_p^{PUASW}(A_v) \leq \frac{q_h^2}{2^l} + \frac{(q_s+q_e)^2}{(q-1)} + \frac{2q_s}{2^l}$$

B. Informal security analysis

The informal security features of the proposed scheme are discussed as follows.

1) User Anonymity: User anonymity property is preserved by not compromising ID_m during the communication. In the registration phase, the patient's ID_i is transmitted over a secure channel. Hence, the adversary cannot get identity by intercepting message during the registration phase. In the login and authentication phase, the adversary cannot get identity from UID_i, L_2 as it is protected by one way hash function. Hence, user anonymity is preserved in the proposed authentication scheme.

2) User untraceability: User untraceability property is satisfied if the attacker cannot guess if the two messages are coming from the same user or not. In the PUASW, all parameters of message $M_1 = \{UID_i, NID_i, L_2, T_u\}$ change due to use of unique random integers n_1 for every session. T_u is the timestamp, which confirms the freshness of the message. Hence, an adversary cannot guess whether two different messages are coming from the same user. All parameters of message M_1 is different for each session. Hence, in the PUASW user untraceablity property is satisfied.

3) Mutual Authentication: The mutual authentication property is satisfied if both parties involved in the authentication process authenticate each other. In the PUASW, the server authenticates a user by comparing $ACK^* \stackrel{?}{=} h(ID_c^* \| SID \| SK)$, where the session key is computed both user and application server. Similarly, AP_n is authenticated by the user U_c by comparing the $R_2^* \stackrel{?}{=} h(SID \| A_n \| SK^*)$. If the condition holds, the application server is valid. Hence, the PUASW achieves mutual authentication.

4) Replay attack: A replay attack occurs when an adversary captures the message and later transmits it. The receiving party considers this message as a fresh message transmitted from a genuine party. In the proposed authentication scheme, the user sends request message M_1 including the timestamp T_u and application server response message M_3 to the user. Even if an attacker intercepts transmitted message M_1 or M_2 and retransmits later, the freshness of message is confirmed by timestamp and the random number n_1 and n_3 respectively. Thus, the PUASW can resist replay attack.

5) Insider Attack: The legitimate user sends the computed password $PW_{i1} = h(PW_c \| b)$ to the RC instead of an original password. As the password is protected by the hash function which is computationally difficult to retrieve the password. Thus, neither registration center nor insider experience the original password.

6) Session key temporary information attack: It may happen that the random number $n1$ and $n3$ are revealed. However, the PUASW can resist the session key temporary information attack as session keys are computed as $SK = T_{n_1}(R_1)$ and $SK^* = T_{n_3}(L_1)$, where R_1 and L_1 are nowhere reveal through the communication. So, compromised of two random numbers will not reveal any session key.

7) Impersonation Attack: In this attack, no adversary can generate a valid login message $\{UID_i, NID_i, L_2, T_u\}$. In the PUASW for login message A_v needs to know ID_c, n_1, n_2 and the master key x, which is an infeasible work for him.

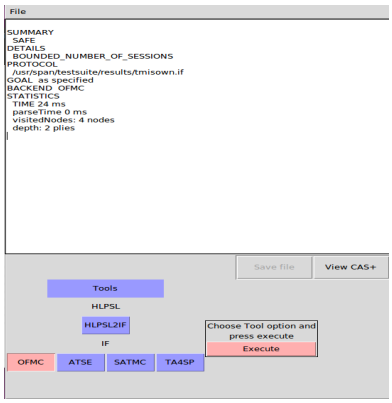
V. Formal security verification using AVISPA tool

The PUASW has been verified using the AVISPA tool, which is considered as one of the power tools to validate the protocol [24], [25]. The tool contains four back ends such as OFMC, CL-Atse, SATMC, and TA4sp. The protocol is written in high-level protocols specification language (HLPSL) and translated into intermediate format (IF). The output of the IF is given to anyone of the back-end and then the results show whether the scheme is safe or unsafe.

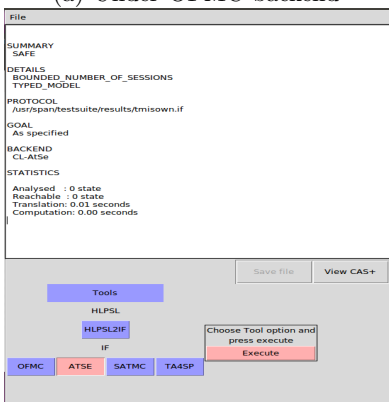
Figure 1a and Figure 1b confirms the simulation of our scheme under two back ends that is OFMC and Atse. The simulation results show the scheme is safe.

VI. Performance Comparison of the proposed scheme

This section represents the comparison of the PUASW with other competent schemes in terms of computational cost, communicational cost and security features. Table IV summarizes the computational and communicational cost of the schemes. Cryptographic hash function (T_{HS}) takes 0.00058s and chaotic map function (T_{CH}) takes 0.02104s for computation [19]. For communicational analysis, we have assumed the length of identity/password/timestamp/nonce is 32 bits, the encryption/decryption is 128 bits, and the output of hash function/chaotic map function is 160 bits. From the table, the communicational cost of the scheme is less than [18] and [19] scheme. Although, the scheme [17] has less communicational cost, however, the scheme has suffered from many attacks.



(a) Under OFMC backend



(b) Under ATSE backend

Fig. 1: Simulation result using AVISPA tool

TABLE IV: Comparison of Computational cost

Scheme	[17]	[18]	[19]	Proposed scheme
Registration	$3T_{HS}$	$3T_{HS}$	$5T_{HS}$	$5T_{HS}$
Login and authentication	$11T_{HS}+6T_{CH}$	$19T_{HS}+6T_{CH}$	$29T_{HS}+6T_{CH}$	$12T_{HS}+4T_{CH}$
Total	$14T_{HS}+6T_{CH} \approx 0.13436s$	$22T_{HS}+6T_{CH} \approx 0.139s$	$34T_{HS}+6T_{CH} \approx 0.145965s$	$17T_{HS}+4T_{CH} \approx 0.09402s$
Communication cost	1152 bits	2752 bits	3232 bits	1472 bits

Table V manifest the security features of PUSAW and other related schemes. It shows that the scheme in [17] and [18] vulnerable to impersonation attack and session key temporary information attack. Besides, the scheme [19] is vulnerable to replay attack and scheme [18] could not withstand insider attack. However, PUASW can resist several well-known attacks.

TABLE V: Comparison of security features

Scheme	AS1	AS2	AS3	AS4	AS5	AS6	AS7
[17]	$\mathcal{Y}$	$\mathcal{Y}$	$\mathcal{Y}$	$\mathcal{Y}$	$\mathcal{Y}$	$\mathcal{N}$	$\mathcal{N}$
[18]	$\mathcal{Y}$	$\mathcal{Y}$	$\mathcal{Y}$	$\mathcal{Y}$	$\mathcal{N}$	$\mathcal{N}$	$\mathcal{N}$
[19]	$\mathcal{Y}$	$\mathcal{Y}$	$\mathcal{Y}$	$\mathcal{N}$	$\mathcal{Y}$	$\mathcal{Y}$	$\mathcal{Y}$
Proposed scheme	$\mathcal{Y}$	$\mathcal{Y}$	$\mathcal{Y}$	$\mathcal{Y}$	$\mathcal{Y}$	$\mathcal{Y}$	$\mathcal{Y}$

$AS1$-User Anonymity, $AS2$- User untracebility, $AS3$- -Mutual Authentication, $AS4$-Replay attack, $AS5$-Insider Attack, $AS6$- session key temporary information attack, $AS7$- Impersonation Attack.
$\mathcal{Y}$- Yes, $\mathcal{N}$- No

VII. Conclusion

In this article, we design a secure chaotic map-based authentication scheme for WBAN which resist several well-known attacks and achieves security features. The formal proof with the ROR model manifests the session key security of the scheme. Moreover, the informal security analysis of the scheme proved that the scheme can withstand several known attacks. High security and less computational and communicational cost show the scheme is suitable for practical applications.

References

[1] L. Lamport, "Password authentication with insecure communication," Communications of the ACM, vol. 24, no. 11, pp. 770–772, 1981.

[2] C.-C. Chang and T.-C. Wu, "Remote password authentication with smart cards," IEE Proceedings E (Computers and Digital Techniques), vol. 138, no. 3, pp. 165–168, 1991.

[3] H.-M. Sun, "An efficient remote use authentication scheme using smart cards," IEEE Transactions on Consumer Electronics, vol. 46, no. 4, pp. 958–961, 2000.

[4] L. Fan, J.-H. Li, and H.-W. Zhu, "An enhancement of timestamp-based password authentication scheme," Computers & Security, vol. 21, no. 7, pp. 665–667, 2002.

[5] C.-W. Lin, C.-S. Tsai, and M.-S. Hwang, "A new strong-password authentication scheme using one-way hash functions," Journal of Computer and Systems Sciences International, vol. 45, no. 4, pp. 623–626, 2006.

[6] L.-H. Li, L.-C. Lin, and M.-S. Hwang, "A remote password authentication scheme for multiserver architecture using neural networks," IEEE Transactions on Neural Networks, vol. 12, no. 6, pp. 1498–1504, 2001.

[7] I.-C. Lin, M.-S. Hwang, and L.-H. Li, "A new remote user authentication scheme for multi-server architecture," Future Generation Computer Systems, vol. 19, no. 1, pp. 13–22, 2003.

[8] W.-S. Juang, "Efficient multi-server password authenticated key agreement using smart cards," IEEE Transactions on Consumer Electronics, vol. 50, no. 1, pp. 251–255, 2004.

[9] E.-J. Yoon and K.-Y. Yoo, "Robust biometrics-based multi-server authentication with key agreement scheme for smart cards on elliptic curve cryptosystem," The Journal of supercomputing, vol. 63, no. 1, pp. 235–255, 2013.

[10] R. Amin, S. Islam, M. K. Khan, A. Karati, D. Giri, and S. Kumari, "A two-factor rsa-based robust authentication system for multiserver environments," Security and Communication Networks, vol. 2017, 2017.

[11] D. Xiao, X. Liao, and K. Wong, "An efficient entire chaos-based scheme for deniable authentication," Chaos, Solitons & Fractals, vol. 23, no. 4, pp. 1327–1331, 2005.

[12] S. Han, "Security of a key agreement protocol based on chaotic maps," Chaos, Solitons & Fractals, vol. 38, no. 3, pp. 764–768, 2008.

[13] E.-J. Yoon and K.-Y. Yoo, "A new key agreement protocol based on chaotic maps," in KES International Symposium on Agent and Multi-Agent Systems: Technologies and Applications. Springer, 2008, pp. 897–906.

[14] X. Guo and J. Zhang, "Secure group key agreement protocol based on chaotic hash," Information Sciences, vol. 180, no. 20, pp. 4069–4074, 2010.
[15] D. He and M. K. Khan, "Cryptanalysis of a key agreement protocol based on chaotic hash." IJESDF, vol. 5, no. 3/4, pp. 172–177, 2013.
[16] W.-J. Tsaur, J.-H. Li, and W.-B. Lee, "An efficient and secure multi-server authentication scheme with key agreement," Journal of Systems and Software, vol. 85, no. 4, pp. 876–882, 2012.
[17] C.-C. Lee, D.-C. Lou, C.-T. Li, and C.-W. Hsu, "An extended chaotic-maps-based protocol with key agreement for multiserver environments," Nonlinear Dynamics, vol. 76, no. 1, pp. 853–866, 2014.
[18] X. Li, J. Niu, S. Kumari, S. H. Islam, F. Wu, M. K. Khan, and A. K. Das, "A novel chaotic maps-based user authentication and key agreement protocol for multi-server environments with provable security," Wireless Personal Communications, vol. 89, no. 2, pp. 569–597, 2016.
[19] A. Irshad, S. A. Chaudhry, Q. Xie, X. Li, M. S. Farash, S. Kumari, and F. Wu, "An enhanced and provably secure chaotic map-based authenticated key agreement in multi-server architecture," Arabian Journal for Science and Engineering, vol. 43, no. 2, pp. 811–828, 2018.
[20] D. Dolev and A. Yao, "On the security of public key protocols," IEEE Transactions on information theory, vol. 29, no. 2, pp. 198–208, 1983.
[21] P. Kocher, J. Jaffe, and B. Jun, "Differential power analysis," in Advances in cryptology—CRYPTO'99. Springer, 1999, pp. 789–789.
[22] T. S. Messerges, E. A. Dabbish, and R. H. Sloan, "Examining smart-card security under the threat of power analysis attacks," IEEE transactions on computers, vol. 51, no. 5, pp. 541–552, 2002.
[23] M. Abdalla, P.-A. Fouque, and D. Pointcheval, "Password-based authenticated key exchange in the three-party setting," in Public Key Cryptography - PKC 2005, S. Vaudenay, Ed. Berlin, Heidelberg: Springer Berlin Heidelberg, 2005, pp. 65–84.
[24] A. Armando, D. Basin, Y. Boichut, Y. Chevalier, L. Compagna, J. Cuéllar, P. H. Drielsma, P.-C. Héam, O. Kouchnarenko, J. Mantovani et al., "The avispa tool for the automated validation of internet security protocols and applications," in International Conference on Computer Aided Verification. Springer, 2005, pp. 281–285.
[25] A. A. V. of Internet Security Protocols and Applications. (2015) http://www.avispa-project.org/.