Construction of a chaotic map-based authentication protocol for online telemedicine services

Meena Sanjay Babulal 1*

Abstract
Upgraded network technology yields a new interface "telecare medicine information systems" in short TMIS, between patient and server. But, it can be observed that these services generally insecure as the information being transmitted over a public channel. Chaotic map plays highly important role in designing an authentication protocol, and a good candidate to ensure the efficiency. Although few of the protocols needs low computation cost, but they cannot establish an anonymous communication. Keeping these facts in mind, one needs to construct a chaotic map based authentication scheme for a TMIS. Moreover, we have done performance analysis of related schemes to show the advantage of our work.

Keywords

AMS Subject Classification
81Q50, 37D99, 11G20, 11T55, 94A62.

1 Department of Mathematics, University of Rajasthan, Jaipur-302004, India.
*Corresponding author: sanjaymeena.iitb@gmail.com

Article History: Received 13 September 2020; Accepted 20 November 2020

1. Introduction
Telemedicine is an emerging sector in medical field not only for convenience of patients but also for doctors. The pandemic Covid 19 make it necessary that initial interaction of health workers and patient should be done through online platforms. Telemedicine is not a new concept, it is traces back to the mid-20th century, when radio has been used to provide health advice on ships. For hospitals the first uses were for psychiatric services in the 1950’s via a closed circuit television connection. Telemedical care has expanded over the past 30 years to include stroke, mental health and chronic deceased patients such as diabetes, asthma or heart failure.

With the increasing growth of mobile and software related applications, Telemedicine services will potentially provide essential healthcare coverage for rural and remote areas where advanced medical knowledge is unavailable. Since the Internet is really an open network with multiple possible safety holes, careful attention and precautions must be exercised to ensure medical and patient data protection.
been going on and completed. Nowadays, both computer science and information technology are promoting healthcare services (see Fig. 1, 2) around the world. Employing robots and smart sensors in surgeries, machine learning and artificial intelligence techniques in medical diagnosis, pervasive computing systems for any time, anywhere and medical care, and distributed systems for processing big data of medical are just a few examples of computer based applications in healthcare.

One of the highly important system TMIS is very useful in various healthcare services remote areas such as medical monitoring, consultation, and other health conscious and necessarily convenient services, which are the current demand in various healthcare sectors. These services facilitates private health related support to the concerned patient at their home. Everyday, both engineer and researcher are developing innovative ideas to develop advanced healthcare services.

Therefore, the people can be facilitated with health-services through their smart phone, i-pads, and other electronic devices, where privacy of the users and their access to the service plays highly important role. Therefore, Wu et al. [9] designed an advanced authentication protocol to benefit the healthcare services. In 2012, Wei et al. [8] observes [9] is vulnerable to two-factor authentication. Therefore, a new design comes in demand under two-factor authentication. Zhu [11] shows password guessing in [8] and introduced an improved scheme, but he did not think about anonymous communication. Chen et al. [3] introduced highly demanding authentication protocol preserving anonymous communication (in TMIS services). Lin et al. [6] observes weakness in [3] as identity can be revealed using the dictionary and password guessing can be done with stolen smart card. Therefore, he invented an anonymous protocol to emove most of the existing attacks. Cao and Zhai [2] also found that [3] is not resistant against identity guessing and password with the information of smart card. These schemes [2, 6, 11] vulnerable to input verifying condition due to which these schemes cannot efficiently distinguish incorrect input. Two highly important attributes are anonymity and unlinkability, which are missing in [8, 9, 11, 31].

To improve the efficiency and security chaotic cryptosystems has been developed. In 2013, Guo et al. [13] introduced a new authentication scheme using chaotic-cryptosystem, but Hao et al. [14] claims that both user’s traceability and two secret keys are the main issues. Therefore, Hao et al. introduced a new scheme better than [13]. Jiang et al. [15] carefully analysed the weakness in [14] i.e. stolen smart card attack.

Li et al. [20] designed an advanced chaotic map-based authentication protocol used to healthcare services, Madhusudhan et al. [19] analysed various attacks such as password guessing, and impersonation. Jiang et al. [27] designed an improved TMIS, but it exchanges three messages to establish secure session key. Wu et al. [28] introduced Rfid based authentication and Radhakrishnan et al. [18] designed an advanced TMIS, but it is also vulnerable to password, identity guessing and stolen smart card attacks. Zhang et al. [24] introduced demanding advanced authentication protocol for healthcare TMIS, but it is also vulnerable to various attacks such as identity, password and replay. Madhusudhan et al. [19] designed a robust protocol used to telecare medical information system, it can be observed their scheme is suffering identity, password, impersonation, and stolen smart card attacks. Recently, Dharminder et al. [33] proposed a new construction of RSA based authentication in authorized access to healthcare services, but it uses costly modulo exponentiation.

From Table 2, one can observe security analysis of chaotic based authentication in TMIS, where notation √ stands for "yes", and × for "not". Table 1, 2 shows that existing schemes for TMIS faces various vulnerabilities. Therefore, we need a new TMIS using chaotic-system with the following features:

- Both user login and password change must be efficient.
- Linkability, identity and password guessing resistant.
- Security and efficiency.
Therefore, a new scheme has been designed along with both security and efficiency using chaos theory based cryptosystem. This scheme essentially analysed in random Oracle as well as using BAN logic. Another important contribution of the scheme to resist session key violation introduced by Bergamo et al. [32].

2. Preliminaries

Notations and basic definitions are discussed to analyze some properties of Chaos theory and the scheme. The brief description of the used notations are given as in Table 3.

2.1 Chebyshev Chaotic Mapping

Chaotic mapping possesses an advanced structure (as shown in Figure 3) in nonlinear dynamics, and its security and pseudo randomness. We are basically trying to illustrate some of basic definition and necessary properties in brief could be found in [16].

- **Definition 1** Chebyshev proposed a polynomial essentially in variable ”x” described as

\[
T_\kappa(x) : (−\infty, +\infty) \to [−1, +1] \text{ of positive degree } \kappa, \text{ whereas } T_\kappa(x) = \cos(\kappa \arccos(x)) \text{ and the recurrence}
\]

![Chebyshev polynomials](image)
<table>
<thead>
<tr>
<th>Security attributes/Schemes</th>
<th>[23]</th>
<th>[14]</th>
<th>[21]</th>
<th>[22]</th>
<th>[24]</th>
<th>[18]</th>
<th>[20]</th>
<th>[19]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Anonymous</td>
<td>√</td>
<td>√</td>
<td>×</td>
<td>√</td>
<td>×</td>
<td>×</td>
<td>×</td>
<td>×</td>
</tr>
<tr>
<td>Linkability</td>
<td>×</td>
<td>√</td>
<td>√</td>
<td>√</td>
<td>√</td>
<td>×</td>
<td>×</td>
<td>×</td>
</tr>
<tr>
<td>Insider-attack</td>
<td>√</td>
<td>×</td>
<td>√</td>
<td>√</td>
<td>√</td>
<td>×</td>
<td>×</td>
<td>×</td>
</tr>
<tr>
<td>Password-guess</td>
<td>√</td>
<td>√</td>
<td>√</td>
<td>×</td>
<td>×</td>
<td>×</td>
<td>×</td>
<td>×</td>
</tr>
<tr>
<td>Stolen-card</td>
<td>√</td>
<td>√</td>
<td>√</td>
<td>√</td>
<td>√</td>
<td>√</td>
<td>√</td>
<td>√</td>
</tr>
<tr>
<td>Impersonation-user</td>
<td>√</td>
<td>×</td>
<td>√</td>
<td>×</td>
<td>√</td>
<td>×</td>
<td>×</td>
<td>×</td>
</tr>
<tr>
<td>Impersonation-server</td>
<td>√</td>
<td>√</td>
<td>√</td>
<td>√</td>
<td>√</td>
<td>√</td>
<td>×</td>
<td>×</td>
</tr>
<tr>
<td>Replay</td>
<td>√</td>
<td>×</td>
<td>√</td>
<td>×</td>
<td>√</td>
<td>√</td>
<td>√</td>
<td>√</td>
</tr>
<tr>
<td>Key-agreement</td>
<td>√</td>
<td>√</td>
<td>√</td>
<td>√</td>
<td>√</td>
<td>√</td>
<td>√</td>
<td>√</td>
</tr>
<tr>
<td>Key-verification</td>
<td>√</td>
<td>×</td>
<td>√</td>
<td>×</td>
<td>√</td>
<td>√</td>
<td>√</td>
<td>√</td>
</tr>
</tbody>
</table>

Table 2. Security comparison of chaotic map-based authentication schemes for TMIS

<table>
<thead>
<tr>
<th>Notation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>$U_i$</td>
<td>User-i</td>
</tr>
<tr>
<td>$S_j$</td>
<td>Server-j</td>
</tr>
<tr>
<td>$\mathcal{A}$</td>
<td>Adversary</td>
</tr>
<tr>
<td>$S_C$</td>
<td>Smart Card</td>
</tr>
<tr>
<td>$ID_i$</td>
<td>Identity of $U_i$</td>
</tr>
<tr>
<td>TMIS</td>
<td>Telecare medicine information system</td>
</tr>
<tr>
<td>$PW_i$</td>
<td>Password of $U_i$</td>
</tr>
<tr>
<td>$s$</td>
<td>Secret value of $S_j$</td>
</tr>
<tr>
<td>$h(\cdot)$</td>
<td>A collision resistant hashing</td>
</tr>
<tr>
<td>$h_b(\cdot)$</td>
<td>A collision resistant biometric-hashing</td>
</tr>
<tr>
<td>$\oplus$</td>
<td>Bitwise XOR</td>
</tr>
<tr>
<td>$</td>
<td></td>
</tr>
</tbody>
</table>

Table 3. Notations and Symbols

$T_k(x)$ is defined as $T_k(x) = (2xT_{k-1}(x) - T_{k-2}(x))$, whereas $x \in (-\infty, +\infty)$ and $T_0(x) = 1, T_1(x) = x$.

- **Definition 2** Discrete Logarithm Problem (DLP) is defined for known $y$ and $x$, it is computationally infeasible to find $u$ such that $T_u(x) = y$.

- **Definition 3** Computational Diffie-Hellman Problem (CDHP) is to find out $T_w(x)$ if one essentially knows $x$, $T_w(x)$ and $T_v(x)$.

2.2 Threat model

In this subsection, we are assuming a threat model as in Figure(1) and notations in Table 3 under the scheme [7], where $\mathcal{A}$ possesses some computational resources and smart card security in both password and chaotic based authentication schemes.

- The user essentially chooses an arbitrary pseudo-random password from the dictionary. Server generates its own concerned private key and inserts essential values in the smart card.

- $\mathcal{A}, U_i$ and $S_j$ interact via executing oracle queries that allow $\mathcal{A}$ to simulates an attack on the authentication protocol.

- Communication channel is controlled and managed by the $\mathcal{A}$, where interception, modification, sending again the messages and divert the message are possible.

- The $\mathcal{A}$ may also steal the information stored in smart card.

3. Proposed authentication protocol under chaotic mapping

We have proposed an advanced chaotic mapping based authentication protocol, that has been divided into four phases, (1) registration, (2) login, (3) authentication and (4) password update respectively.

3.1 Registration-Phase

$U_i$ executes the registration process (see Fig 5) along with $S_j$ via a private channel as described in the following lines.

- $U_i$ selects $ID_i$, $PW_i$, and imprints his own biometric $H_i = h_b(\text{biometric})$, then he does the computation $A_i = h(ID_i||PW_i||H_i)$ and transmits $\{ID_i, A_i\}$ to corresponding $S_j$.

- After getting the information $\{ID_i, A_i\}$, $S_j$ stores $ID_i$ of $U_i$ and it computes $x = h(ID_i||s)$, where ”$s$” is the secret key of server $S_j$. Now, $S_j$ chooses arbitrary $n_i \in Z_p^*$, then

![Communication model in TMIS](image-url)
it computes $T_x(I_D_i||n_i)$ that corresponds to $U_i$. Furthermore, it computes the value $B_i = T_x(I_D_i||n_i) \oplus A_i$.

- $S_j$ delivers a hidden information $\{h(\cdot), B_i, n_i\}$ stored in the smart-card that is given to $U_i$ via a private channel.

- Finally, $U_i$ does the computations $T_x(I_D_i||n_i) = B_i \oplus A_i$, $D_i = h(T_x(I_D_i||n_i)||P_W||I_D_i||H_i)$, $N_i = n_i \oplus A_i$ and store $D_i$ with the corresponding values $\{h(\cdot), B_i, D_i, N_i\}$.

- $S_j$ performs the computation $Z_i = h(S_k_i||I_D_i||T_2)$ and transmits the information $\{Z_i, T_2\}$ to corresponding $U_i$.

- After receiving the information $\{Z_i, T_2\}$ from $S_j$, then $U_i$ confirms the time stamp $T_2$ is valid or not, then he computes $Z_i' = h(S_k_i||I_D_i||T_2)$, and proceeds the verification $Z_i' = Z_i$, and establishes a session key $S_k = T_x(I_D_i||n_i)$.

### 3.4 Password update phase

$U_i$ can have access to update the password executing the following steps:

- $U_i$ inputs the card $S_{Cl_i}, I_D_i$ and $P_W_i$. Furthermore, he imprints biometric, then computes $H_i = h_b(\text{biometric})$ and $A'_i = h(I_D_i||P_W_i||H_i)$. Using $A'_i$ the $S_{Cl_i}$ obtains $T_x(I_D_i||n_i)' = A'_i \oplus B_i$ and $D_i' = h(T_x(I_D_i||n_i)'||P_W_i||I_D_i)$, and proceeds for the verification $D_i' = D_i$.

- $U_i$ inputs $P_{Wd_{\text{new}}}$ then $S_{Cl_i}$ and proceeds the computation $A_{\text{new}} = h(I_D_i||P_{W_{\text{new}}}| |H_i)$, $D_{\text{new}} = h(T_x(I_D_i||n_i)| |P_{W_{\text{new}}}| |I_D_i||H_i)$, $B_{\text{new}}^{\text{new}} = A_{\text{new}} \oplus T_x(I_D_i||n_i)$ and updates the values $B_i, D_i$ with $B_i^{\text{new}}, D_i^{\text{new}}$.

### 4. Security Analysis

#### 4.1 Security proof in Random oracle

Initially, we need to describe a model $\mathcal{P}$ to validate the security of the proposed scheme and then will implement the proposed protocol under random oracle (RO).

**Security-Model** Let $C_i \in \{U_i, S_j\}$ be an $i$th instance, where $\mathcal{A}$ is an adversary who controls communication between $U_i$ and $S_j$.

**Extract:** This phase actually permits $\mathcal{A}$ to retrieve corresponding private key of $U_i$.

**Send($M, C_i$):** This permits $\mathcal{A}$ to send arbitrary message $M$, then Oracle sends the final output to $\mathcal{A}$.

**Hash($m$):** $\mathcal{A}$ sends arbitrary $m$ to the hashing $H(\cdot)$, it chooses arbitrary $s \in Z_p$ and stores $(m, s)$ and returns the arbitrary number.

**Reveal($C_i$):** This permits $\mathcal{A}$ to get the knowledge about $CK_{ij}$ (session key), when an Oracle gets a reveal query.

**Corrupt($C_i$):** This allows $\mathcal{A}$ to disturb and to make a corruption to the party $C_i$ and it obtains key of the corrupted party $C_i$.

**Test($C_i$):** This permits Oracle to receive a message from $\mathcal{A}$, then Oracle guesses a $c \in \{0, 1\}$. If $c = 0$, then it sends an arbitrary random number, and If $c = 1$, then oracle return a session key $CK_{ij}$.
Let $Succ(\mathcal{A})$ successfully guess the value of a bit $c$, which is chosen from the corrupt query phase. The advantage held by the $\mathcal{A}$ against the authentication protocol is defined as:

$$Adv_{\mathcal{A}}(p(k) = |2Pr[Succ(\mathcal{A})] - 1|$$

Security analysis for the proposed scheme achieves mutual authentication in RO.

**Chaotic based assumption:** For random oracle construction on chaotic map, basic primitive as follows: From generation algorithm $Gen(1^n) = p$, where $p$ is prime of length $n$ and $x \in Z^n_p$ is secret key of $S_j$.

For probabilistic polynomial time adversary $\mathcal{A}$, $\exists$ a negligible function $neg(n)$ such that:

$$Pr[Gen(1^n) \rightarrow p, x, r, T_x(r) \leftarrow Z_p^* : \mathcal{A}(1^n, p, T_x(r)) \rightarrow x] = neg(n)$$

**Collision resistance attack (CRA) algorithm:** If $\mathcal{A}$ finds a collision for one way hash function $h(\cdot)$, we have

$$Adv_{\mathcal{A}} = Pr[(x, x') \leftarrow R : x \neq x' \text{ and } h(x) = h(x')]$$

**Theorem 4.1.** Let $H$ be a random Oracle and $\mathcal{A}$ be an adversary, then we will model an algorithm $B$ who solve CAA problem using subroutine $\mathcal{A}$.

**Proof** Initially, $B$ receives an instance $C_i$, $F_i$, $H_i$ and then $B$ tries to solve the CAA instance via computing $X_i$ and $r^*$ and checking the conditions $h(r^* || ID_i || PW_i) = ?D_i$ or $F_i = h(T_i(ID_i || N_i) || Sk_B || T_1)$. Finally, the information $H$, $\omega$, $p$, $T$, $Gen(\cdot)$ is made public for $B$, who can approach to $\mathcal{A}$.

*H hash query:* When $\mathcal{A}$ submits a query which corresponds to $ID_i$, then $B$ carefully first verifies $ID_i$ in $H_{ij}$ list. If present in the list named $H_{ij}$, then $B$ returns exactly the value $h_{13}$, otherwise it computes the value $h_{a_i}$ and puts in the list along with $(ID_i, h_a)$. and sends back $h_a$ to $\mathcal{A}$.

*Extract:* $\mathcal{A}$ submits an advanced query on $ID_i$, then $B$ receives the corresponding query and proceeds for the verification $H_i(ID_i) \in \{C_i, F_i, T_i\}$. If verification does not hold, then $B$ terminates the process. After following this procedure, $B$ goes for the verification $ID_i \in H_{ij}$, if present, then it gives response, else calculates $X_i = T_i(ID_i || N_i)$ and $D_i = h(T_i(ID_i || N_i) || PW_i || ID_i || T_1)$ returns to $\mathcal{A}$.

*Send-queries:* Send phase is described as below $U_i$ login and sends a message $< (C_i, F_i, T_i) >$ to $S_j$, and $S_j$ responds $(H_i, T_2)$. This phase is described with the help of a game played between $U_i$ and $S_j$, respectively.

1. $\mathcal{A}$ sends a query, then $B$ returns a login message to $\mathcal{A}$.

2. To get login in to $S_j$, $\mathcal{A}$ submits polynomial times send queries, then $B$ does a computation corresponding to $i^{th}$ query as $X_i = T_i(ID_i || N_i)$ and $V_i = h(T_i(ID_i || N_i) || PW_i || N_i)$ responds to $\mathcal{A}$.

3. $\mathcal{A}$ submits $(V_i, U_i)$, then $B$ verifies first whether $H(ID_i) \in H_{ij}$ or not. If it is present, then $B$ returns a
Figure 7. Computation cost comparison

Figure 8. Communication cost comparison

failure, otherwise it terminates the query $E_{el}$. Further, $B$ computes $M_1 = T_y(ID_i || N_i)$, and $Sk_u = T_1 T_y(ID_i || N_i)$ for arbitrary $z, y$ and $F_i = h(T_y(ID_i || N_i) || Sk_u || T_1)$ and returns the output to $A$. 

2133
If table 4. An analysis of performance with recent chaotic
happened. Thus, 

\[ A \] sends are legal when event \( \text{time of computation hashing, extracts and} \]

information about the private key, then it sends a duplicate

Otherwise, \( B \) is able to solve the CAA problem. As

\( H(ID) \notin H_j \), \( Z = h(Y) \notin (x_1, x_2, \ldots) \), and \( \delta \) be chance of

success of \( B \), and \( \epsilon \) be the chance of breaching the scheme

respectively. Then, each of the queries hashing, extracts and

sends are legal when event \( E_x \), \( E_{x_1} \), \( E_{x_2} \) exist. So, \( B \) takes

the help of \( \mathcal{A} \) to break CAA problem, if none of \( E_x \), \( E_{x_1} \), \( E_{x_2} \)

happened. Thus,

\[
Pr[\neg E_x \land \neg E_{x_1} \land \neg E_{x_2}] = \left( \frac{q_{E_x}}{q_H} \right)^{q_E+q_1} \left( \frac{q_H - q_{E_x}}{q_H} \right)^{q_E+q_1}
\]

Therefore, \( B \) is successful with advantage

\[
\delta \geq \left( \epsilon - \frac{1}{2k} \right) \left( \frac{q_{E_x}}{q_H} \right)^{q_E+q_1} \left( \frac{q_H - q_{E_x}}{q_H} \right)^{q_E+q_1}
\]

So, the algorithm \( B \) has gain the advantage as above, therefore

if \( \mathcal{A} \) gets success in breaking the protocol, then \( B \) can use subroutine \( \mathcal{A} \) to break the proposed scheme.

5. Performance Analysis

The performance analysis describes efficiency of the proposed
protocol as compared to the related protocols in Table 4, where \( t_h \approx 0.00205s \), \( t_{sym} \approx 0.0087s \), \( t_c \approx 0.02102s \)
and \( t_m \approx 0.06307s \) denote the time of computation hashing, symmetric-encryption, chaotic-map operation, multiplication in \( Z_p^* \) respectively.

<table>
<thead>
<tr>
<th>Schemes</th>
<th>User-computation</th>
<th>Server-computation</th>
<th>Messages</th>
</tr>
</thead>
<tbody>
<tr>
<td>Liu et al.’s [21]</td>
<td>4t_h + 2t_c</td>
<td>5t_h + 2t_sym + 3t_c</td>
<td>3</td>
</tr>
<tr>
<td>Jiang et al.’s [15]</td>
<td>2t_h + t_sym + t_c</td>
<td>2t_h + 2t_sym + 3t_c</td>
<td>2</td>
</tr>
<tr>
<td>Hao et al.’s [14]</td>
<td>2t_c + 3t_h + 2t_c</td>
<td>2t_c + 3t_sym + 2t_h</td>
<td>2</td>
</tr>
<tr>
<td>Lee’s [23]</td>
<td>2t_c + t_h</td>
<td>2t_c + 8t_h</td>
<td>2</td>
</tr>
<tr>
<td>Zhang et al.’s [24]</td>
<td>6t_h + 2t_c</td>
<td>4t_h + 1t_c + 2t_sym</td>
<td>3</td>
</tr>
<tr>
<td>Madhusudhan et al.’s [19]</td>
<td>7t_h + 2t_c</td>
<td>7t_h + 2t_c</td>
<td>2</td>
</tr>
<tr>
<td>Li et al.’s [1]</td>
<td>7t_c + 2t_c</td>
<td>7t_c + 2t_c</td>
<td>2</td>
</tr>
<tr>
<td>Proposed</td>
<td>3t_c + 2t_h</td>
<td>2t_c + 2t_h</td>
<td>2</td>
</tr>
</tbody>
</table>

Table 4. An analysis of performance with recent chaotic map-based authentication protocols

This section presents a performance analysis of authentication schemes [1, 14, 15, 19, 21, 23, 24]. All the schemes related to telecare medicine services highly depend on operations which requires low computation and limited storage. Both efficiency and performance has been compared to the related protocols in Table 4, whereas the cost of each operation is computed via running the experiment on intel Pentium – 4 processor with 1024 MB ram as in [5, 34] along with computation cost illustrated in Figure 7.

Moreover, Liu et al.’s scheme [21] takes \( 4t_h + + 2t_c \) for user, \( 5t_h + 2t_c \) for server, Jiang et al.’s scheme [15] takes \( 2t_h + t_sym + t_c \) for user, \( 2t_h + 2t_sym + 3t_c \) for server, Hao et al.’s scheme [14] takes \( 2t_c + 3t_h + 2t_sym \) for user, \( 2t_c + 3t_sym + 2t_h \) for server, Lee et al.’s scheme [23] takes \( 2t_c + 7t_h \) for user, \( 2t_c + 8t_h \) for server, Zhang et al.’s scheme [24] takes \( 6t_h + 2t_c \) for user, \( 4t_h + t_c + 2t_sym \) for server, Madhusudhan et al.’s scheme [19] takes \( 7t_h + 2t_c \) for user, \( 3t_h + 2t_c \) for server, Li et al.’s scheme [1] takes \( 7t_h + 2t_c \) for user, \( 7t_h + 2t_c \) for server, where the presented protocol executes \( 3t_c + 2t_h \) for user, \( 2t_c + 2t_h \) for server respectively.

In this paper, we adopt the communication cost of hash function, chaotic map and time stamp are given the 160-bit output, and symmetric encryption is 256 bits, where total communication overhead is given in Figure 8. The computation comparison of proposed schemes is shown in Figure 8.

6. Conclusion

This article describes the security of recently presented chaotic map based authentication in the random Oracle. The proposed protocol successfully removes the existing vulnerabilities and observes that how a poor verification invites various attacks. Furthermore, it can be observed that the presented design ensures session key verification after just two messages exchange.

References

[6] Lin, H.Y.: On the security of a dynamic id-based authen-
Construction of a chaotic map-based authentication protocol for online telemedicine services — 2135/2136


Dharminder, Dharminder and Mishra, Dheerendra and Li, Xiong Construction of RSA-Based Authentication Scheme in Authorized Access to Healthcare Services
Construction of a chaotic map-based authentication protocol for online telemedicine services — 2136/2136


**********
ISSN(P):2319 – 3786
Malaya Journal of Matematik
ISSN(O):2321 – 5666
**********