



An efficient authentication with key agreement procedure using Mittag–Leffler–Chebyshev summation chaotic map under the multi-server architecture

Chandrashekhar Meshram¹ · Rabha W. Ibrahim² · Sarita Gajbhiye Meshram³ · Sajjad Shaukat Jamal⁴ · Agbotiname Lucky Imoize^{5,6}

Accepted: 19 August 2021 / Published online: 14 September 2021

© The Author(s), under exclusive licence to Springer Science+Business Media, LLC, part of Springer Nature 2021

Abstract

The recent technological advancement and rapid development of computer networks have increased the popularity of remote password authentication protocols. Toward this end, the emphasis has shifted to protocols that apply to smart cards-empowered multi-server environments. In order to defend against the replay attack, these protocols usually depend on the nonce or timestamp. In this paper, an efficient Mittag–Leffler–Chebyshev Summation Chaotic Map (MLCSCM)-enabled multi-server authentication protocol with the key agreement is proposed and generalized to address this peculiarity in multi-server-oriented applications. The security proof and efficiency analysis of the presented MLCSCM authenticated key agreement protocol is rigorously derived and validated. Compared to the recently published literature, the proposed protocol presents high efficiency with unique features, and it is highly resistant to sophisticated attacks and achieves perfect forward secrecy.

Keywords Mittag–Leffler–Chebyshev Summation Chaotic Map (MLCSCM) · Computer networks · Mutual authentication · Multi-server architecture · Key exchange · Smart card

1 Introduction

The widespread adoption of the Internet globally is attributed to its numerous benefits and usefulness in government parastatals, non-governmental agencies, educational institutions, smart cities, industries, private sectors, and others [1]. There are various applications in which clients can access various services from multiple networks remotely, such as healthcare, banking, smart grid, smart agriculture, home

✉ Chandrashekhar Meshram
csmeshram84pdf@gmail.com

Extended author information available on the last page of the article

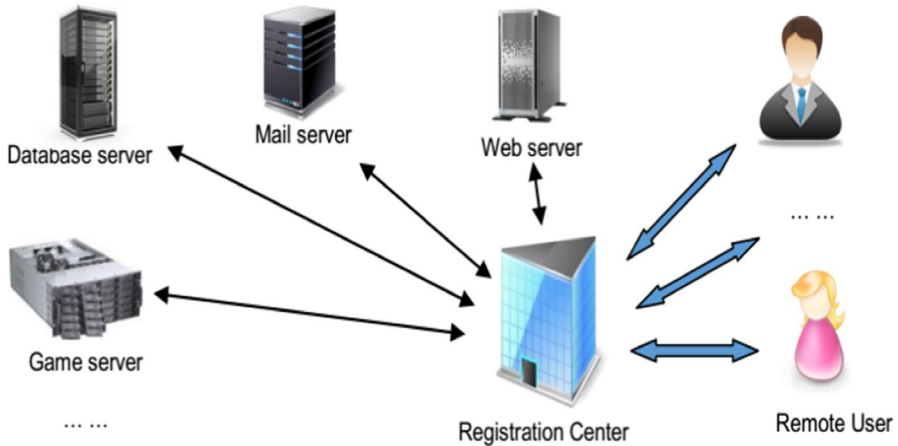


Fig. 1 The multi-server communication architecture [9]

automation, and more. Consequently, this has contributed significantly to the complexity of the communication network and the associated peripherals [2]. Therefore, it becomes very challenging to satisfy the needs of all clients on the server simultaneously. To this end, a multi-server-based system (see Fig. 1) that allows applicants to access services at any time without interruption is presented. The database, mail, Web, game, and remote servers are interconnected via highly secured channels to the registration center.

The smart card, a small device with low power consumption, has become increasingly popular as the electronics industry expands. Notably, password authentication protocols were created to secure sensitive data stored on servers from being accessible to any hostile intruder. It is worth mentioning that these protocols have gained widespread acceptance due to their ease of implementation, low cost, and user-friendliness [3–6]. Currently, the traditional password authentication protocols are no longer optimal and lack the capabilities to support modern multi-server systems used in the design of computer networks [6]. Recently, the Kerberos method [7], one of the most promising password authentication protocols, was reported. The scheme is explicitly built for multi-server-based environments. However, the Kerberos method is not entirely secure from sophisticated intrusion. The system has consistently shown vulnerability to password guessing attacks, especially when the client chooses a weak password. This implies that participants must use strong cryptographic secrets for the Kerberos system to operate efficiently.

Furthermore, Tsai [8] projected a multi-server authentication procedure that utilizes a one-way hash function. It is remarkable to note that this authentication system does not require the use of a verification table. However, the Tsai procedure is exposed to a man-in-the-middle attack. In particular, the Tsai procedure depends on the timestamp or nonce to prevent replay attacks. Unfortunately, the introduction of clock synchronization is cost-prohibitive. To address the concerns mentioned above,

Tsaur et al. [6] suggested a self-checked timestamp procedure in 2012. In this case, the timestamp creator can easily verify the timestamp for a suitable application.

In 2014, Lee et al. [10] proposed an elaborate authentication procedure, which uses chaotic maps to defend against multiple attacks. However, the scheme shows limited security against denial-of-service and session key attacks. Additionally, the scheme requires that the server and the client exchange messages up to three times to create a service link. This could be time-consuming and inefficient. In another related study, Banerjee et al. [11] reported that earlier authentication procedures had smart card loss and were susceptible to user impersonation attacks. To this end, the authors proposed a smart card-enabled anonymous authentication procedure to protect the system against security threats such as impersonation, forward secrecy, smart card loss, and insider attacks. In 2016, Sun et al. [12] discovered several flaws in [11] and recommended a dynamic identity authentication procedure to defend against multiple attacks. However, the procedure due to Sun et al. [12] is not entirely secure against multiple attacks such as offline password guessing, replay, and smart card loss. Recently, Li et al. [13] discovered some security issues in the scheme reported by [10] and suggested a robust key agreement procedure, which is based on chaotic theory, to improve the system in [10].

Similarly, Irshad et al. [14] discovered some security flaws in [13]. They also proposed an advanced framework to combat the established security issues. However, Irshad et al. [14] procedures necessitate huge communication overhead, high computational complexity, and ample storage requirements. Additionally, Jangirala et al. [15] found several security vulnerabilities in the previous method [13] and proposed a more reliable authentication procedure to address the issues. They also claimed that while the procedure in [15] is immune to multiple attacks, it shows minimal performance when tested against several security metrics.

Ying and Nayak [16] recently proposed a remote user authentication procedure to improve the performance and security of multi-server architectures. Interestingly, the authentication procedure uses self-certified public key cryptography, which is claimed to secure against specific attacks. However, this procedure shows vulnerability to smart card loss, impersonation, replay, session key disclosure, password guessing, and insider attack. Furthermore, it necessitates more computing resources, which is highly undesirable in communication networks. Chaos theory-based cryptography has attracted huge research interests in recent years, leading to some notable accomplishments, such as signature techniques [17–20], authentication [21–23], encryption techniques [24, 25], mobile healthcare [26], hash functions [27], privacy preservation [28], and blockchain-enabled certificateless schemes [29]. Table 1 gives a summary of the associated works.

The preceding works have not adequately addressed authentication with the key agreement under the multi-server communication, which is crucial to the design and development of security systems for multi-servers of the future. Therefore, the need to fill this gap is not out of place. To this end, this paper proposes an efficient multi-server authentication procedure with key agreement, leveraging the Mittag–Leffler–Chebyshev Summation Chaotic Map (MLCSCM). In our design, clients will be able to interact anonymously with the server using the new

Table 1 Summary of associated works

References	Key contributions	Limitations
Tsai [8]	An effective multi-server authentication procedure using a one-way hash function is proposed. The procedure does not need to store any verification table in the registration center and server	The security of employing a one-way hash function as a foundation for authentication purposes has not been evaluated
Tsaur et al. [6]	A protected multi-server authentication procedure with the key agreement was proposed, and a self-verified Timestamp method was developed to enable the smart-card-based authentication procedure	The procedure is vulnerable to both the privileged insider attack and the known-plaintext attack. Additionally, disturbance in clock synchronization and vulnerability to the man-in-the-middle attack was not fully addressed. The procedure's communication costs must be reduced, and the procedure's performance must be improved
Lee et al. [10]	An authentication procedure with key agreement using extended chaotic maps for multi-server environments is proposed	The procedure appears to incur high computational costs that need to be reduced. Furthermore, the procedure is ineffective at detecting unauthorized logins and lacks a password-changing option
Banerjee et al. [11]	The work presents an anonymous multi-server remote user authentication procedure using smart card	Practical implementation of the scheme is not presented
Sun et al. [12]	The work examined Banerjee et al.'s procedure [11] and found that the procedure is vulnerable to user impersonation attacks and offline password guessing attacks. The authors projected an enhanced procedure to remove the security vulnerability	The scheme is not entirely secure against multiple attacks such as offline password guessing, replay, and smart card loss
Li et al. [13]	The work presents authentication with key agreement protocol using chaotic map for multi-server applications	The establishment of a security investigated model for authentication and key agreement protocol using chaotic is required
Irshad et al. [14]	The work presents a provably secure authenticated key agreement using chaotic map in multi-server setting to improve Li et al.'s protocol [13] at minimum possible computation cost	Forward secrecy and two-factor security need to be examined thoroughly
Jangirala et al. [15]	The work proposed an efficient authentication procedure in multi-server setting that can preserve all the original merits of Shunmuganathan et al.'s procedure [30] and withstand the possible known attacks	Practical implementation of the procedure is missing, and the future extension of the procedure is not elaborated
Ying and Nayak [16]	The work presents a lightweight and untraceable authentication protocol for multi-server-based 5G wireless networks. The authors employed a self-certified public key cryptography using elliptic curve to authenticate the justification of servers and users	The scheme appears to be computationally complex with a very high communication cost of 212.96 ms

Mittag–Leffler–Chebyshev Summation Chaotic Map (MLCSCM)-based multi-server authentication procedure that can provide mutual authentication between clients and servers.

The main contributions of this paper are outlined as follows. First, we presented a robust description of the Chebyshev chaotic maps, Mittag–Leffler–Chebyshev Summation (MLCS), MLCS chaotic maps, extended MLCS, and their properties. Second, we proposed an efficient authentication with key agreement procedure using Mittag–Leffler–Chebyshev summation chaotic map under the multi-server architecture. Furthermore, we present the formal authentication proof based on the BAN logic to demonstrate the strengths of the proposed scheme. Additionally, we provide a check to confirm that the presented protocol supports mutual authentication, user anonymity, and perfect forward secrecy. Also, we put the presented protocol to the test against several attacks, including the replay attack, privileged insider attack, Bergamo et al.’s attack, and the known-plaintext attack. Finally, we provide security examinations to show that our proposed procedure can provide comprehensive security assurance at a low computation cost while demonstrating very high efficiency due to the utilization of the Mittag–Leffler–Chebyshev summation chaotic maps and hash functions.

The rest of this paper is laid out as follows. The definitions of the Mittag–Leffler–Chebyshev Summation (MLCS), MLCS chaotic maps, extended MLCS, and its properties are introduced in Sect. 2. Section 3 presents the proposed MLCSCM-based multi-server authentication with key agreement procedure. Section 4 offers solid proof to demonstrate the effectiveness and efficiency of the proposed procedure based on BAN logic. Section 5 shows the results of the proposed security system, and Sect. 6 presents the performance analysis of the procedure. Finally, the conclusion to the paper is given in Sect. 7.

2 Background and materials

In this segment, we provide some background knowledge of the presented protocol. This comprises the Chebyshev chaotic maps, Mittag–Leffler–Chebyshev Summation (MLCS), MLCS chaotic maps, extended MLCS, and its properties. The notations used in the proposed procedure are defined in Table 2.

2.1 Chebyshev chaotic maps

The following are some fundamental ideas regarding the Chebyshev polynomial [31]. The Chebyshev polynomial $T_n(\tau)$ is a degree n polynomial in τ . Let n be an integer, and τ be a variable with a range of values between $[-1, 1]$. The Chebyshev polynomial (CP) $T_n(\tau) : [-1, 1] \rightarrow [-1, 1]$ is given as

$$T_n(\tau) = \cos(n \cdot \cos^{-1}(\tau)).$$

The recurrence relation of the CP is defined as

$$T_n(\tau) = 2\tau T_{n-1}(\tau) - T_{n-2}(\tau), \quad n \geq 2$$

where $T_0(\tau) = 1$ and $T_1(\tau) = \tau$.

The functions $\cos(\tau)$ and $\cos^{-1}(\tau)$ are trigonometric [32]. They are referred to as $\cos : \mathcal{R} \rightarrow [-1, 1]$ and $\cos^{-1} : [-1, 1] \rightarrow [0, \pi]$, individually.

2.2 Mittag–Leffler–Chebyshev summation (MLCS)

The Mittag–Leffler function $E_{\alpha,\beta}$ is a special function, which indicates two parameters α and β . It may be demarcated by the resulting series when the real parameter α is strictly positive [33]:

$$E_{\alpha,\beta}(\chi) = \sum_{k=0}^{\infty} \frac{\chi^k}{\Gamma(\alpha k + \beta)} \tag{1}$$

where $\Gamma(\cdot)$ is the gamma function. When $\beta = 1$, it is abbreviated as $E_{\alpha}(\chi) = E_{\alpha,1}(\chi)$. For $\alpha = 0$, the series above generates the Taylor growth of the geometric series and given accordingly, as follows:

$$E_{0,\beta}(\chi) = \frac{1}{\Gamma(\beta)} \frac{1}{1 - \chi} \tag{2}$$

Definition 1 For two power series in χ .

$$\phi(\chi) = \sum_{n=0}^{\infty} \phi_n \chi^n, \tag{3}$$

the transform $\mathcal{B}_{\alpha} \phi$ of ϕ by

$$\mathcal{B}_{\alpha} \phi(\chi) \equiv \sum_{n=0}^{\infty} \frac{\phi_n}{\Gamma(1 + \alpha n)} \chi^n \tag{4}$$

Then, the Mittag–Leffler sum (MLS) of ϕ is given by $\lim_{\alpha \rightarrow 0} \mathcal{B}_{\alpha} \phi(\chi)$.

2.3 MLCS chaotic maps

Definition 2 By using the MLS, we define a transform and call it Mittag–Leffler–Chebyshev Summation (MLCS) [34, 35], and by using the summation formula of Chebyshev polynomials.

$$\mathbb{T}_{\tau}(\chi) := \sum_{n=0}^{\infty} T_n(\tau) \chi^n = \frac{1 - \tau}{1 - 2\chi\tau + \chi^2} \tag{5}$$

Table 2 The notations used in the proposed procedure

Notation	Description
C_i	The i th client
S_j	The j th server
\mathcal{RC}	The registration center
pwd_i	The i th client's password
id_i	The i th client's identity
\mathcal{B}	The private key shared among \mathcal{RC} and S_j
ρ_i	The service period of S_j for C_i
q_1	A large prime number
α	A random real number
w, \varkappa	The private keys of \mathcal{RC}
\mathcal{Y}	The random number selected by \mathcal{RC}
\mathcal{W}	The public key of \mathcal{RC} , where $\mathcal{W} \equiv \mathcal{B}_\alpha \mathbb{T}_\tau(\mathcal{Y}) \pmod{q_1}$
η_i, η_j	Two random integers
$\mathcal{H}(\cdot)$	A secure one-way hash function
\oplus	The exclusive-or operation (XOR)
\parallel	The concatenation operation
$s\kappa$	Session key

Note that

$$T_n(\tau) = n \sum_{k=0}^n (-2)^k \frac{(n+k-1)!}{(n-k)!(2k)!} (1-\tau)^k, \quad n > 0 \tag{6}$$

where

$$\begin{aligned} T_0(\tau) &= 1 \\ T_1(\tau) &= \tau \\ T_{n+1}(\tau) &= 2\tau T_n(\tau) - T_{n-1}(\tau), \end{aligned}$$

as follows:

$$\mathcal{B}_\alpha \mathbb{T}_\tau(\chi) \equiv \sum_{n=0}^{\infty} \frac{T_n(\tau)}{\Gamma(1+\alpha n)} \chi^n \tag{7}$$

satisfying

$$\mathcal{B}_\alpha \mathbb{T}_\tau(\chi) \equiv \sum_{n=0}^{\infty} \frac{[2\tau T_{n-1}(\tau) - T_{n-2}(\tau)]}{\Gamma(1+\alpha n)} \chi^n. \tag{8}$$

Figure 2 displays the dynamic plot of the suggested MLCS.

The chaotic assets and the semigroup assets are the two main assets of Chebyshev polynomials [36].

The chaotic assets The CP map, defined as $T_n(\tau) : [-1, 1] \rightarrow [-1, 1]$ with degree $n > 1$, is a chaotic map with its invariant density function $f^*(\tau) = 1/(\pi\sqrt{1-\tau^2})$ for positive Lyapunov exponent $\lambda = \ln n > 0$.

The semigroup assets The possessions of what is called semigroup satisfy the subsequent equalities:

$$\begin{aligned} T_n(T_l(\tau)) &= \cos(n\cos^{(-1)}(\cos(l\cos^{(-1)}(\tau)))) \\ &= \cos(nl\cos^{(-1)}(\tau)) \\ &= T_{ln}(\tau) \\ &= T_l(T_n(\tau)), \end{aligned}$$

where n and l are positive integers and $\tau \in [-1, 1]$.

The Chebyshev polynomial presents two problems [37], both of which are thought to be hard to solve in polynomial time:

- (1) The discrete log (DL) assignment is to catch the integer w with the end aim $T_n(\tau) = Y$ given two exponents Y and τ .
- (2) Because of three exponents $z, T_n(\tau)$, and $T_l(\tau)$, the Diffie–Hellman problem (DHP) assignment is to measure the $T_m(\tau)$ element.

2.4 The extended MLCS

Zhang [38] presented that the above semigroup assets hold for CPs characterized by $(-\infty, +\infty)$ interval, which can enhance the property, and takes after:

$$T_n(\tau) = (2\tau T_{n-1}(\tau) - T_{n-2}(\tau)) \pmod{\varphi_1}, \tag{9}$$

where $n \geq 2, \tau \in (-\infty, +\infty)$, and φ_1 is a large prime number.

Clearly,

$$T_n(T_l(\tau)) \equiv T_{nl}(\tau) \equiv T_l(T_n(\tau)) \pmod{\varphi_1} \tag{10}$$

Therefore, the semigroup assets quite hold, and the improved MLCS polynomials too commute under composition. By using the convolution methodology for all $\tau \in (-\infty, +\infty)$, we have the following observation.

Theorem 2.1. Consider the MLCS $\mathcal{B}_\alpha \mathbb{T}_\tau(\chi)$. Then, the recurrent relation is

$$(\mathcal{B}_\alpha \mathbb{T}_\tau(\chi))^m = (\mathcal{B}_\alpha \mathbb{T}_\tau^{m-1}(\mathcal{B}_\alpha \mathbb{T}_\tau(\chi))) = (\mathcal{B}_\alpha \mathbb{T}_\tau^{m-2}(\mathcal{B}_\alpha \mathbb{T}_\tau(\chi)^2))$$

and the semigroup relation is

$$(\mathcal{B}_\alpha \mathbb{T}_\tau^k(\mathcal{B}_\alpha \mathbb{T}_\tau^m(\chi))) = (\mathcal{B}_\alpha \mathbb{T}_\tau^{mk}(\chi)) = (\mathcal{B}_\alpha \mathbb{T}_\tau^m(\mathcal{B}_\alpha \mathbb{T}_\tau^k(\chi))).$$

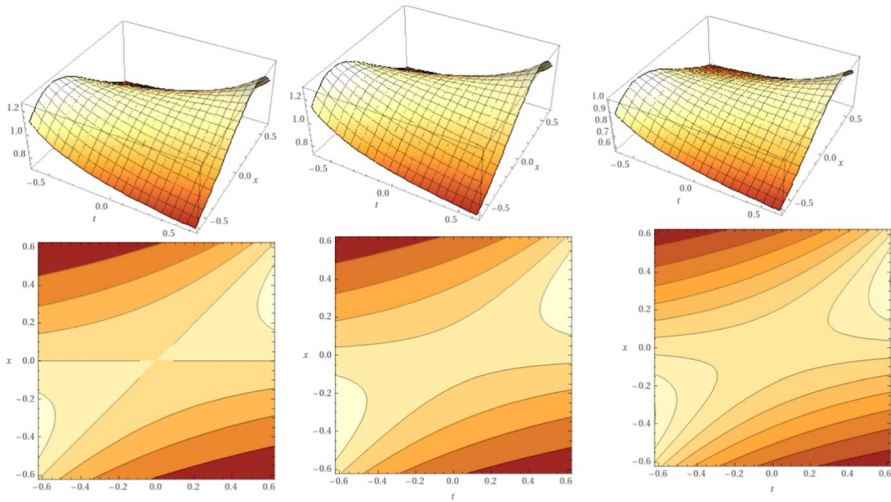


Fig. 2 The 3D plot of $\mathcal{B}_\alpha \mathbb{T}_\tau(\chi)$, when $\alpha=0.25, 0.5, 0.75$, respectively

Proof The first part is directly obtained by operating m -times MLCS $\mathcal{B}_\alpha \mathbb{T}_\tau(\chi)$.
 For the second part, we have

$$\begin{aligned}
 (\mathcal{B}_\alpha \mathbb{T}_\tau^k(\mathcal{B}_\alpha \mathbb{T}_\tau^m(\chi))) &= \left(\mathcal{B}_\alpha \mathbb{T}_\tau \underbrace{(\dots \dots (\mathcal{B}_\alpha \mathbb{T}_\tau))}_{k\text{-times}} \right) (\mathcal{B}_\alpha \mathbb{T}_\tau^m(\chi)) \\
 &= \left(\mathcal{B}_\alpha \mathbb{T}_\tau \underbrace{(\dots \dots (\mathcal{B}_\alpha \mathbb{T}_\tau))}_{k\text{-times}} \right) \left(\mathcal{B}_\alpha \mathbb{T}_\tau \underbrace{(\dots \dots (\mathcal{B}_\alpha \mathbb{T}_\tau))}_{m\text{-times}} \right) \\
 &= \left(\mathcal{B}_\alpha \mathbb{T}_\tau \underbrace{(\dots \dots (\mathcal{B}_\alpha \mathbb{T}_\tau))}_{k+1\text{-times}} \right) \left(\mathcal{B}_\alpha \mathbb{T}_\tau \underbrace{(\dots \dots (\mathcal{B}_\alpha \mathbb{T}_\tau))}_{m-1\text{-times}} \right) \\
 &\quad \vdots \\
 &= (\mathcal{B}_\alpha \mathbb{T}_\tau^{mk}(\chi)).
 \end{aligned}$$

Similarly, we have

$$\begin{aligned}
 (\mathcal{B}_\alpha \mathbb{T}_\tau^m (\mathcal{B}_\alpha \mathbb{T}_\tau^k (\chi))) &= \left(\mathcal{B}_\alpha \mathbb{T}_\tau (\underbrace{\dots \dots (\mathcal{B}_\alpha \mathbb{T}_\tau)}_{m\text{-times}}) \right) (\mathcal{B}_\alpha \mathbb{T}_\tau^m (\chi)) \\
 &= \left(\mathcal{B}_\alpha \mathbb{T}_\tau (\underbrace{\dots \dots (\mathcal{B}_\alpha \mathbb{T}_\tau)}_{m\text{-times}}) \right) \left(\mathcal{B}_\alpha \mathbb{T}_\tau (\underbrace{\dots \dots (\mathcal{B}_\alpha \mathbb{T}_\tau)}_{k\text{-times}}) \right) \\
 &= \left(\mathcal{B}_\alpha \mathbb{T}_\tau (\underbrace{\dots \dots (\mathcal{B}_\alpha \mathbb{T}_\tau)}_{m+1\text{-times}}) \right) \left(\mathcal{B}_\alpha \mathbb{T}_\tau (\underbrace{\dots \dots (\mathcal{B}_\alpha \mathbb{T}_\tau)}_{k-1\text{-times}}) \right) \\
 &\quad \vdots \\
 &= (\mathcal{B}_\alpha \mathbb{T}_\tau^{mk} (\chi)).
 \end{aligned}$$

3 The proposed MLCSCM-based multi-server authentication with key agreement procedure

To solve the security problems, we present a multi-server authentication protocol based on the MLCS chaotic maps. The registration center \mathcal{RC} chooses a \mathcal{Y} random number, two random integers (ω, \varkappa) , random real number α , and a private key $\mathcal{B} = \mathcal{h}(\omega \parallel \varkappa)$ to be exchanged among \mathcal{RC} and S_j in our protocol, and then computes $\mathcal{W} \equiv \mathcal{B}_\alpha \mathbb{T}_\tau(\mathcal{Y}) \pmod{q_1}$. The master secret keys (ω, \varkappa) are kept secret by \mathcal{RC} , and \mathcal{B} is sent to S_j through a secure channel. The registration and session key agreement and login phases are described in detail and illustrated in Figs. 3 and 4.

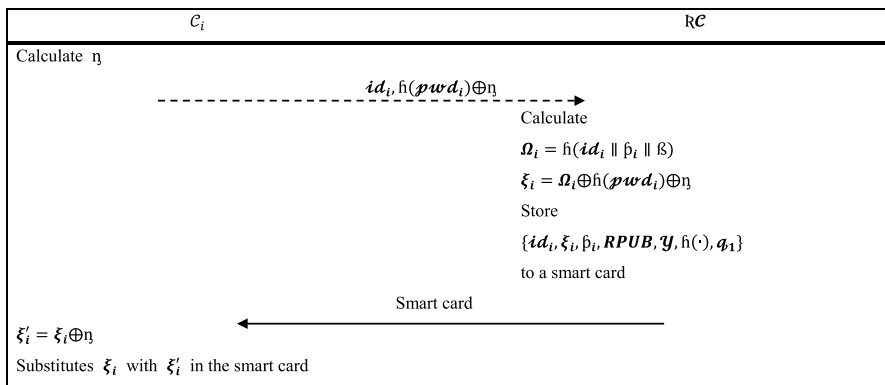


Fig. 3 Registration stage of the presented procedure

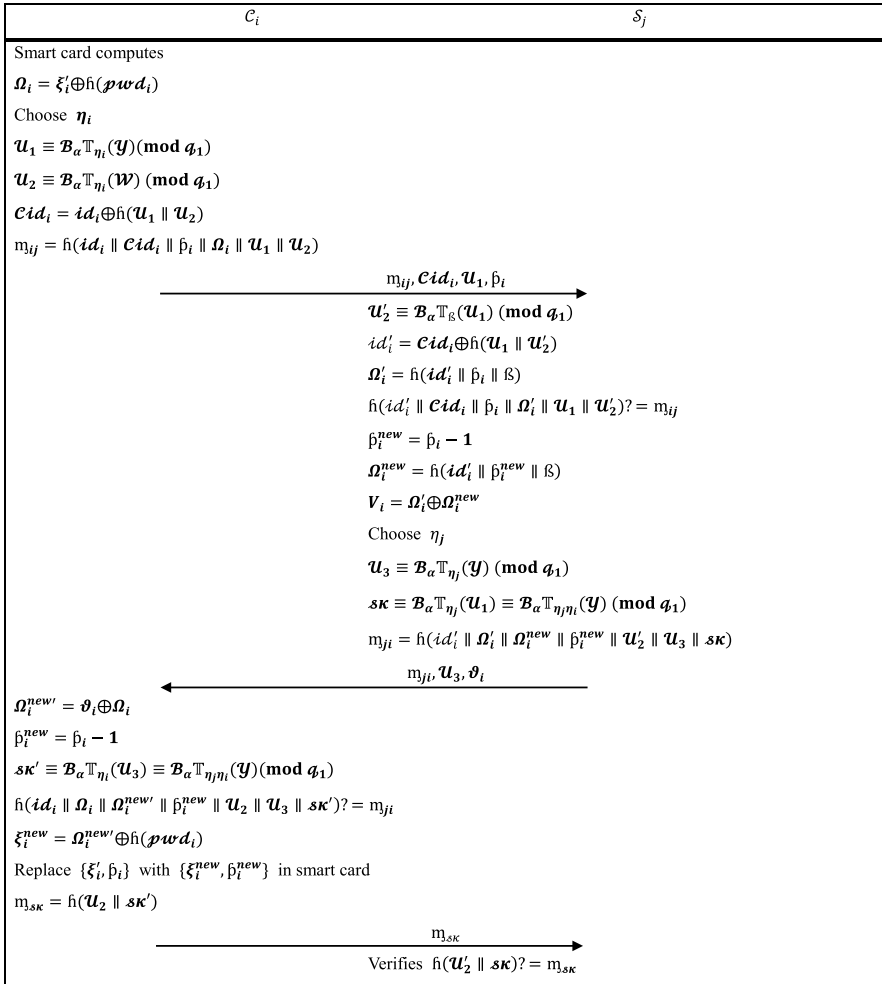


Fig. 4 Login and session key agreement stage of the presented procedure

3.1 Registration stage

When client \mathcal{C}_i wants to use a service given by $\mathcal{S} = \{\mathcal{S}_1, \mathcal{S}_2, \dots, \mathcal{S}_r\}$, \mathcal{C}_i first chooses their id_i identity, pwd_i password, and a random number \mathcal{N} , and then sends $\{id_i, \mathfrak{h}(pwd_i) \oplus \mathcal{N}\}$ to \mathcal{RC} through a secure channel for registration. \mathcal{RC} will take the following actions after getting the message, as represented in Fig. 3:

- (1) Calculate \mathcal{C}_i 's secret info $\Omega_i = \mathfrak{h}(id_i \parallel \rho_i \parallel \mathcal{B})$ and $\xi_i = \Omega_i \oplus \mathfrak{h}(pwd_i) \oplus \mathcal{N}$.
- (2) Store $id_i, \xi_i, \rho_i, RPUB, \mathcal{Y}, \mathfrak{h}(\cdot)$ and q_1 on the memory of a smart card and give it to \mathcal{C}_i .
- (3) \mathcal{C}_i calculates $\xi'_i = \xi_i \oplus \mathcal{N}$ and substitutes ξ_i with ξ'_i in the smart card.

3.2 Login and session key agreement stage

When client C_i wants to log in to the S_j server during this process, they first insert their smart card into a card reader and enter the password $\rho w d_i$. The smart card and S_j will execute the following steps, as shown in Fig. 4:

- (1) The smart card calculates $\Omega_i = \xi_i' \oplus \mathcal{h}(\rho w d_i)$ first and then chooses a η_i random integer. The following is then calculated:

$$U_1 \equiv \mathcal{B}_\alpha \mathbb{T}_{\eta_i}(\mathcal{J})(\text{mod } \varphi_1), \quad U_2 \equiv \mathcal{B}_\alpha \mathbb{T}_{\eta_i}(\mathcal{W})(\text{mod } \varphi_1),$$

$$C i d_i = i d_i \oplus \mathcal{h}(U_1 \parallel U_2), \quad m_{ij} = \mathcal{h}(i d_i \parallel C i d_i \parallel \rho_i \parallel \Omega_i \parallel U_1 \parallel U_2).$$

A $\{m_{ij}, C i d_i, U_1, \rho_i\}$ message is created and conveyed to S_j .

- (2) Subsequently, getting $\{m_{ij}, C i d_i, U_1, \rho_i\}$, S_j tests the equation $\mathcal{h}(i d_i' \parallel C i d_i \parallel \rho_i \parallel \Omega_i' \parallel U_1 \parallel U_2) = m_{ij}$ by calculating the succeeding:

$$U_2 \equiv \mathcal{B}_\alpha \mathbb{T}_B(U_1)(\text{mod } \varphi_1), \quad i d_i' = C i d_i \oplus \mathcal{h}(U_1 \parallel U_2), \quad \Omega_i' = \mathcal{h}(i d_i' \parallel \rho_i \parallel \mathcal{B}).$$

S_j rejects the login request if the equation above does not hold; otherwise, the service duration ρ_i is reviewed again to see if it has expired. S_j will terminate the service given to C_i if ρ_i expires; otherwise, S_j will update the service duration ρ_i with $\rho_i^{new} = \rho_i - 1$, and then calculate the new private info $\Omega_i^{new} = \mathcal{h}(i d_i' \parallel \rho_i^{new} \parallel \mathcal{B})$ for C_i . To cover Ω_i^{new} , S_j calculates $\vartheta_i = \Omega_i' \oplus \Omega_i^{new}$ and picks a random integer η_j . S_j calculates $U_3 \equiv \mathcal{B}_\alpha \mathbb{T}_{\eta_j}(\mathcal{J})(\text{mod } \varphi_1)$ and the session key $\mathcal{JK} \equiv \mathcal{B}_\alpha \mathbb{T}_{\eta_j}(U_1) \equiv \mathcal{B}_\alpha \mathbb{T}_{\eta_j \eta_i}(\mathcal{J})(\text{mod } \varphi_1)$ using the random integer η_j . S_j then calculates $m_{ji} = \mathcal{h}(i d_i' \parallel \Omega_i' \parallel \Omega_i^{new} \parallel \rho_i^{new} \parallel U_2 \parallel U_3 \parallel \mathcal{JK})$ and sends the $\{m_{ji}, U_3, \vartheta_i\}$ message to C_i .

- (3) The smart card verifies the equation $\mathcal{h}(i d_i \parallel \Omega_i \parallel \Omega_i^{new'} \parallel \rho_i^{new} \parallel U_2 \parallel U_3 \parallel \mathcal{JK}') = m_{ji}$ when it receives the $\{m_{ji}, U_3, \vartheta_i\}$ message from S_j , by working out the following equation:

$$\Omega_i^{new'} = \vartheta_i \oplus \Omega_i, \quad \rho_i^{new} = \rho_i - 1, \quad \mathcal{JK}' \equiv \mathcal{B}_\alpha \mathbb{T}_{\eta_i}(U_3) \equiv \mathcal{B}_\alpha \mathbb{T}_{\eta_j \eta_i}(\mathcal{J})(\text{mod } \varphi_1).$$

If the above equation holds, the smart card computes $\xi_i^{new} = \Omega_i^{new'} \oplus \mathcal{h}(\rho w d_i)$ and replaces $\{\xi_i', \rho_i\}$ with $\{\xi_i^{new}, \rho_i^{new}\}$; otherwise, the relation is dropped. $m_{\mathcal{JK}} = \mathcal{h}(U_2 \parallel \mathcal{JK}')$ is then calculated by the smart card and transmitted to the server S_j .

S_j confirms the validity of the (\mathcal{JK}) session key by checking whether the equation $\mathcal{h}(U_2' \parallel \mathcal{JK}) = m_{\mathcal{JK}}$ holds after receiving the message $\{m_{\mathcal{JK}}\}$ from C_i . Both C_i and S_j can use \mathcal{JK} to secure a subsequent contact session if the session key is right. If this relation is not maintained, it will be terminated.

4 The formal authentication proof based on BAN logic

Burrows et al. [39] proposed the BAN logic used to verify the accuracy of the proposed procedure. In the area of cryptographic procedure analysis, the BAN logic is one of the most recognized methods. The logic allows for a quick, easy, and formal study of cryptographic protocols [39, 40]. Here, we describe the basic notations, priorities, and expectations before using the BAN logic. The following are the specifics.

5 Notations

First of all, we examine the BAN logic’s syntax. We define \mathfrak{A} , \mathfrak{B} , as participants and ρ as a formula, respectively. Additionally, we use some examples to examine the BAN logic’s syntax and notations [39, 40].

- $\mathfrak{A} \equiv \rho$: \mathfrak{A} assumes that ρ is right.
- $\mathfrak{A} \triangleleft \rho$: ρ is seen or carried by \mathfrak{A} .
- $\mathfrak{A} \equiv \mathfrak{B}$: \mathfrak{A} trusts on \mathfrak{B} ’s activities, e.g., $\mathfrak{A} \equiv \mathfrak{B} \triangleleft \rho$ means that \mathfrak{A} trusts on \mathfrak{B} hold ρ .
- $\mathfrak{A} \Rightarrow \rho$: ρ is fully under \mathfrak{A} ’s influence. This is a term that can be used to refer to a certificate authority.
- $\mathfrak{A} | \sim \rho$: \mathfrak{A} once said ρ .
- $\#(\rho)$: ρ is new, which means it occurred recently, or ρ is a nonce.
- $\mathfrak{A} \stackrel{\rho}{\leftrightarrow} \mathfrak{B}$: ρ is a private key or secret info communal among \mathfrak{A} and \mathfrak{B} .
- ρ \mathfrak{A} and ρ^{-1} : \mathfrak{A} has a ρ public key and a ρ^{-1} private key.
- $\overrightarrow{\{m\}}_{\rho}$: Plain text m is encrypted by ρ .
- (ρ, ζ) : ρ or ζ is one portion of the formula (ρ, ζ) .
- $\frac{Rule1}{Rule2}$: We can conclude *Rule2* from *Rule1*, e.g., $\frac{\mathfrak{A} \text{ creates random } \rho}{\mathfrak{A} \equiv \#(\rho)}$ means that \mathfrak{A} produces ρ , so \mathfrak{A} trusts ρ is fresh.

As shown in Fig. 4, we use the BAN logic to turn our proposed procedure into an idealized form. The idealized versions of the messages are as follows:

- M1. $C_i \rightarrow S_j : \mathfrak{h}(C_i d_i, \mathcal{P}_i, \Omega_i, \{\mathcal{Y}\}_{\eta_i}, \{\mathcal{Y}\}_{\mathcal{B}_{\eta_i}}), C_i d_i, \{\mathcal{Y}\}_{\eta_i}, \mathcal{P}_i$
- M2. $S_j \rightarrow C_i : \mathfrak{h}(\Omega'_i, \Omega_i^{new}, \mathcal{P}_i^{new}, \{\mathcal{Y}\}_{\mathcal{B}_{\eta_i}}, \{\mathcal{Y}\}_{\eta_j}, C_i \stackrel{\mathcal{JK}}{\leftrightarrow} S_j), \{\mathcal{Y}\}_{\eta_j}, \vartheta_i$
- M3. $C_i \rightarrow S_j : \mathfrak{h}(\{\mathcal{Y}\}_{\eta_i}, C_i \stackrel{\mathcal{JK}}{\leftrightarrow} S_j)$

5.1 Objectives

The objectives of our projected procedure are then specified in BAN logic syntax. Participants in the proposed protocol include legal user C_i , legal user C_j , and trusted authority TA . C_i trusts C_j , where C_j is a legal user, and C_j believes C_i is a legal user are

the four targets of our procedure. In the BAN logic language, the goals of the proposed procedure are represented as formula O1–O4.

- O1. $C_i | \equiv C_i \stackrel{JK}{\leftrightarrow} S_j$
- O2. $S_j | \equiv C_i \stackrel{JK}{\leftrightarrow} S_j$
- O3. $C_i | \equiv S_j | \equiv C_i \stackrel{JK}{\leftrightarrow} S_j$
- O4. $S_j | \equiv C_i | \equiv C_i \stackrel{JK}{\leftrightarrow} S_j$

5.2 Expectations

We made the following expectations to analyze our procedure using the BAN logic:

- A1. $C_i | \equiv \#(\eta_i)$
- A2. $S_j | \equiv \#(\eta_B)$
- A3. $C_i | \equiv S_j \stackrel{B}{\leftrightarrow} \mathcal{RC}$
- A4. $S_j | \equiv S_j \leftrightarrow \mathcal{RC}_B$
- A5. $C_i | \equiv S_j | \equiv S_j \stackrel{B}{\leftrightarrow} \mathcal{RC}$
- A6. $S_j | \equiv C_i | \equiv S_j \stackrel{JK}{\leftrightarrow} \mathcal{RC}$
- A7. $C_i | \equiv S_j | \Rightarrow C_i \leftrightarrow S_j$
- A8. $S_j | \equiv C_i \leftrightarrow S_j$

5.3 Verification

The correctness of the presented protocol is demonstrated in this subsection by evaluating the idealized version of our protocol using the expectations defined in Sect. 4.3 and the BAN logic rules. The following are the key steps in the proof:

C_i picks random η_i

- V1. $C_i | \equiv \eta_i$
- V2. $C_i | \equiv \#(\eta_i)$

Message 1: $C_i \rightarrow S_j : \mathcal{R}(\{\mathcal{Y}\}_{\eta_i}, \{\mathcal{Y}\}_{B \cdot \eta_i}), \{\mathcal{Y}\}_{\eta_i}$

- V3. $S_j \triangleleft \mathcal{R}(\{\mathcal{Y}\}_{\eta_i}, \{\mathcal{Y}\}_{B \cdot \eta_i}), \{\mathcal{Y}\}_{\eta_i}$
- V4. $\frac{S_j \triangleleft \mathcal{R}(\{\mathcal{Y}\}_{\eta_i}, \{\mathcal{Y}\}_{B \cdot \eta_i}), \{\mathcal{Y}\}_{\eta_i}}{S_j | \equiv C_i \sim \{\mathcal{Y}\}_{\eta_i}}$

S_j picks random η_j

- V5. $S_j | \equiv \eta_j$
- V6. $S_j | \equiv \#(\eta_j)$

S_j calculates the session key $C_i \stackrel{JK}{\leftrightarrow} S_j = \{\mathcal{Y}\}_{\eta_i \cdot \eta_j}$
 Message 2: $S_j \rightarrow C_i : \mathcal{R}(\{\mathcal{Y}\}_{B \cdot \eta_i}, \{\mathcal{Y}\}_{\eta_j}, C_i \leftrightarrow S_j), \{\mathcal{Y}\}_{\eta_j}$

- V7.
$$\frac{C_i \triangleleft \mathcal{R} \left(\{\mathcal{Y}\}_{B \cdot \eta_i}, \{\mathcal{Y}\}_{\eta_j}, C_i \stackrel{\delta K}{\leftrightarrow} S_j \right), \{\mathcal{Y}\}_{\eta_j}}{C_i \triangleleft \eta_i, C_i \triangleleft \{\mathcal{Y}\}_{\eta_j}}$$
- V8.
$$\frac{C_i \triangleleft C_i \stackrel{\delta K}{\leftrightarrow} S_j}{C_i \triangleleft \mathcal{R} \left(\{\mathcal{Y}\}_{B \cdot \eta_i}, \{\mathcal{Y}\}_{\eta_j}, C_i \stackrel{\delta K}{\leftrightarrow} S_j \right), \{\mathcal{Y}\}_{\eta_j}, S_j | \equiv C_i | \sim \{\mathcal{Y}\}_{\eta_i}}$$
- V9.
$$\frac{C_i | \equiv S_j | \sim \left(\{\mathcal{Y}\}_{B \cdot \eta_i}, \{\mathcal{Y}\}_{\eta_j}, C_i \stackrel{\delta K}{\leftrightarrow} S_j \right)}{C_i | \equiv \#(\eta_i), C_i | \equiv S_j | \sim \left(\{\mathcal{Y}\}_{B \cdot \eta_i}, \{\mathcal{Y}\}_{\eta_j}, C_i \stackrel{\delta K}{\leftrightarrow} S_j \right)}$$
- V10.
$$\frac{C_i | \equiv S_j | \equiv \left(\{\mathcal{Y}\}_{\eta_i}, \{\mathcal{Y}\}_{\eta_j}, C_i \stackrel{\delta K}{\leftrightarrow} S_j \right)}{C_i | \equiv S_j | \equiv \left(\{\mathcal{Y}\}_{\eta_i}, \{\mathcal{Y}\}_{\eta_j}, C_i \stackrel{\delta K}{\leftrightarrow} S_j \right)}$$
- V11.
$$\frac{C_i | \equiv S_j | \equiv C_i \stackrel{\delta K}{\leftrightarrow} S_j}{C_i | \equiv S_j | \Rightarrow C_i \stackrel{\delta K}{\leftrightarrow} S_j, C_i | \equiv S_j | \equiv \left(\{\mathcal{Y}\}_{\eta_i}, \{\mathcal{Y}\}_{\eta_j}, C_i \stackrel{\delta K}{\leftrightarrow} S_j \right)}$$
- V12.
$$\frac{C_i | \equiv S_j | \Rightarrow C_i \stackrel{\delta K}{\leftrightarrow} S_j, C_i | \equiv S_j | \equiv \left(\{\mathcal{Y}\}_{\eta_i}, \{\mathcal{Y}\}_{\eta_j}, C_i \stackrel{\delta K}{\leftrightarrow} S_j \right)}{C_i | \equiv C_i \stackrel{\delta K}{\leftrightarrow} S_j}$$

Message 3: $C_i \rightarrow S_j : s \left(\{\mathcal{Y}\}_{\eta_i}, C_i \stackrel{\delta K}{\leftrightarrow} S_j \right)$

- V13.
$$\frac{S_j \triangleleft \mathcal{R} \left(\{\mathcal{Y}\}_{\eta_i}, C_i \stackrel{\delta K}{\leftrightarrow} S_j \right)}{S_j | \equiv \#(\eta_i)}$$
- V14.
$$\frac{S_j | \equiv \#(C_i \stackrel{\delta K}{\leftrightarrow} S_j)}{S_j \triangleleft \mathcal{R} \left(\{\mathcal{Y}\}_{\eta_i}, C_i \stackrel{\delta K}{\leftrightarrow} S_j \right), S_j | \equiv C_i \stackrel{\delta K}{\leftrightarrow} S_j}$$
- V15.
$$\frac{S_j | \equiv C_j | \sim C_i \stackrel{\delta K}{\leftrightarrow} S_j}{S_j | \equiv \#(C_i \stackrel{\delta K}{\leftrightarrow} S_j), S_j | \equiv C_j | \sim C_i \stackrel{\delta K}{\leftrightarrow} S_j}$$
- V16.
$$\frac{S_j | \equiv C_j | \equiv C_i \stackrel{\delta K}{\leftrightarrow} S_j}{S_j | \equiv C_i | \equiv C_i \stackrel{\delta K}{\leftrightarrow} S_j}$$

As a result of formulas A8, V16, V11, and V12, we can now be assured that our current procedure can meet the defined objectives.

6 Security examinations

This section provides a check to confirm that the proposed protocol supports mutual authentication, user anonymity, and perfect forward secrecy. Additionally, we put the presented protocol to the test against several attacks, including the replay attack, privileged insider attack, Bergamo et al.’s attack, and the known-plaintext attack.

Hypothesis 1 The presented protocol can accomplish privileged insider attacks.

Proof The user C_i sends $\{id_i, \mathcal{h}(pwd_i) \oplus \mathcal{N}\}$ to the registration center \mathcal{RC} during the registration process of our protocol. Since they cannot acquire the random number \mathcal{N} , a malicious privileged insider would be unable to deduce C_i ’s password and use it to impersonate C_i . As a result, our protocol is unaffected by the privileged insider attack.

Hypothesis 2 The presented protocol can accomplish anonymous client authentication.

Proof If an adversary has eavesdropped on a user C_i 's contact with the server S_j , they can attempt to track C_i 's true identity and obtain confidential information about C_i . The encrypted message $Cid_i = id_i \oplus h(U_1 \parallel U_2)$ protects C_i 's true identity in our presented protocol. The adversary would have to deal with the MLCS chaotic maps-based DLP issue if they tried to extract id_i from Cid_i . To put it another way, we can affirm that our presented protocol gives the user a high degree of anonymity.

Hypothesis 3 The presented protocol can accomplish mutual authentication.

Proof When the server S_j receives the $\{m_{ij}, Cid_i, U_1, \rho_i\}$ message from a user C_i during the session key agreement and login process of our presented procedure, it tests the validity of $h(id_i' \parallel Cid_i \parallel \rho_i \parallel \Omega_i' \parallel U_1 \parallel U_2) = m_{ij}$. S_j considers C_i a legal consumer if the equation holds. Then, $m_{ji} = h(id_i' \parallel \Omega_i' \parallel \Omega_i^{new} \parallel \rho_i^{new} \parallel U_2 \parallel U_3 \parallel \delta\kappa)$ is computed by S_j . $\{m_{ji}, U_3, \vartheta_i\}$ is the message sent to C_i . Similarly, when C_i receives the message $\{m_{ji}, U_3, \vartheta_i\}$ from S_j , it verifies that $h(id_i \parallel \Omega_i \parallel \Omega_i^{new} \parallel \rho_i^{new} \parallel U_2 \parallel U_3 \parallel \delta\kappa') = m_{ji}$ is true. C_i finds S_j a legitimate server if it holds. Since the hidden key \mathcal{B} is known only by the \mathcal{RC} registration center and the server S_j , C_i and S_j store the values $\Omega_i = h(id_i \parallel \rho_i \parallel \mathcal{B})$ and \mathcal{B} , respectively. Finally, the session key sk is produced by both C_i and S_j . This means that the procedure provided perfect mutual authentication among C_i and S_j , making it secure against impersonation attacks.

Hypothesis 4 The presented protocol can counterattack replay attacks.

Proof A replay attack is an arrangement of network attacks in which a legitimate chunk of information is maliciously or fraudulently replicated or delayed. Since the random nonces η_i and η_j ensure the freshness of the messages sent, the replay attack will fail in its attempt to breach our presented protocol. Only S_j (or C_i) can implant the shared mutual session key (sk) and the secret value U_2 in the message m_{ij} (or m_{ji}), except for C_i (or S_j).

Hypothesis 5 The presented protocol can accomplish a known-plaintext attack.

Proof An adversary can be able to easily find \mathcal{Y} , U_1 , and U_3 , but there is no way to extract η_i and η_j from those values in our presented protocol. The explanation for this is that everything has been encrypted using MLCS polynomials, and only the client and the server have access to it. Furthermore, the enhanced MLCS polynomials, i.e., enhanced MLCS chaotic maps, are used in our presented protocol, in the place of any other ordinary Chebyshev polynomials, where the cosine function's periodicity is evaded by extending the interval of \mathcal{y} to $(-\infty, +\infty)$, not to the

indication that the service period ρ_i of any valid client is encrypted by applying $\Omega_i = \mathcal{H}(i \mathcal{A}_i \parallel \rho_i \parallel \mathcal{B})$. As a consequence, we accomplish that the known-plaintext attack would not affect the new protocol provided.

Hypothesis 6 The presented protocol can accomplish perfect forward secrecy.

Proof Even if a long-term key or session key is compromised somehow, the adversary would be unable to extract all other session keys from the cracked one [41, 42] according to perfect forward secrecy. The smart card and server \mathcal{S}_j in our proposed protocol calculate the existing session key $\beta\kappa \equiv \mathcal{B}_\alpha \mathbb{T}_{\eta_i, \eta_j}(\mathcal{Y}) \pmod{\varphi_1}$ using the random numbers η_i and η_j . Even if an adversary knew the current session key $\beta\kappa$, he/she would be unable to use it to calculate any of the other session keys $\beta\kappa \equiv \mathcal{B}_\alpha \mathbb{T}_{\delta_i, \delta_j}(\mathcal{Y}) \pmod{\varphi_1}$ because the random numbers in each contact session are different. This is how we maintain perfect forward confidentiality with our presented protocol.

Hypothesis 7 The presented protocol can accomplish Bergamo et al.’s attack.

Proof The attack due to Bergamo et al. [32] relies on an adversary being able to obtain the related variables \mathcal{Y} , \mathcal{U}_1 , and \mathcal{U}_3 and derive η_i and η_j from them. The adversary may be able to simply find \mathcal{Y} , \mathcal{U}_1 , and \mathcal{U}_3 , but there is no way to extract η_i and η_j from those values in our presented protocol. The elements are encrypted by MLCS chaotic maps and identified by the user and the server for this purpose. Our protocol also uses improved MLCS chaotic maps, which escape the periodicity of the cosine

Table 3 Security attributes comparisons between presented and other associated procedures

Proce- dures → Security attributes ↓	[6]	[4]	[10]	[15]	[14]	[16]	Proposed procedure
S.A1	\mathcal{N}	\mathcal{N}	y	y	y	y	y
S.A2	\mathcal{N}	y	y	y	y	y	y
S.A3	y	y	y	y	y	\mathcal{N}	y
S.A4	\mathcal{N}	y	y	y	y	y	y
S.A5	y	y	y	y	y	y	y
S.A6	\mathcal{N}	\mathcal{N}	y	\mathcal{N}	\mathcal{N}	\mathcal{N}	y
S.A7	\mathcal{N}	\mathcal{N}	y	y	y	\mathcal{N}	y
S.A8	y	\mathcal{N}	\mathcal{N}	y	y	\mathcal{N}	y
S.A9	\mathcal{N}	\mathcal{N}	y	\mathcal{N}	\mathcal{N}	\mathcal{N}	y

S.A1: Privileged insider attack; S.A2: Client anonymity; S.A3: Mutual authentication; S.A4: Impersonation attack; S.A5: Replay attack; S.A6: Known-plaintext attack; S.A7: Perfect forward secrecy; S.A8: Proof of validity; S.A9: Bergamo et al.’s attack

y: Secure; \mathcal{N} : Vulnerable

Table 4 Performance assessments between presented and other related procedures

Procedures	Registration	Login and authentication	Total (ms)
[6]	$4t_h + 2t_s$	$7t_h + 7t_s$	$11t_h + 9t_s \approx 6.4$ ms
[4]	$4t_h + 2t_s$	$7t_h + 7t_s$	$11t_h + 9t_s \approx 6.4$ ms
[10]	$3t_h$	$11t_h + 6t_{ch}$	$14t_h + 6t_{ch} \approx 6.4$ ms
[15]	$8t_h$	$25t_h$	$33t_h \approx 10.56$ ms
[14]	$3t_h$	$29t_h + 6t_{ch}$	$32t_h + 6t_{ch} \approx 12.16$ ms
[16]	$4t_h + 2t_{ec}$	$9t_h + 7t_{ec}$	$13t_h + 9t_{ec} \approx 212.96$ ms
Proposed procedure	$3t_h$	$9t_h + 6t_{ch}$	$12t_h + 6t_{ch} \approx 5.76$ ms

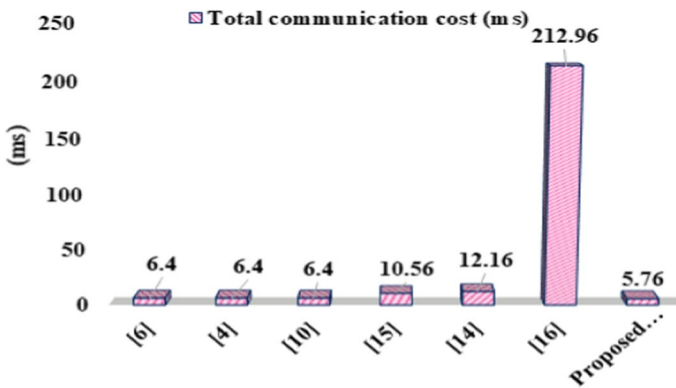


Fig. 5 Total communication cost (ms)

function by extending the y interval to $(-\infty, +\infty)$. As a result, Bergamo et al.’s attack [32] will not impact our presented protocol.

7 Performance discussions

In this segment, we demonstrate the performance of the projected procedure. Table 3 compares the security properties of our proposed procedure and the procedures given by Tsaar et al. [6], Li et al. [4], Lee et al. [10], Jangirala et al. [15], Irshad et al. [14], and Ying and Nayak [16]. Our proposed procedure has an advanced level of protection than the other procedures examined. Additionally, we compared the computational primitives used in the registration process of our proposed procedure, as well as the login and session key agreement stages, to those used in other similar procedures.

In this analysis of contrasts, @@we used the following four notations of time complexity: t_{ch} , t_h , t_s , and t_{ec} reported performance time for a Chebyshev chaotic map operation, a one-way hash function, a symmetric operation, and one elliptic

curve scale multiplication, respectively. The relations among t_h , t_{ch} , t_s , and t_{ec} with respect to t_h ($t_h = 0.32$ ms) have been recognized in several works [17–19, 43]. The following is the relationship and order of computational complexity between the metrics: $t_{ch} \approx t_h$, $t_s \approx t_h$, $t_{ec} \approx 72.5t_h$, and $t_h \approx t_{ch} \approx t_s < t_{ec}$. Table 4 displays the proposed procedures and the main consuming operations of the existing procedures. Comparisons of total computing costs in milliseconds (ms) are also shown in Fig. 5.

By contrast, due to the utilization of the Chebyshev chaotic maps and hash functions, our proposed procedure can provide comprehensive security assurance at a low computation cost while demonstrating very high efficiency.

8 Conclusion

This paper introduces an efficient Mittag–Leffler–Chebyshev Summation Chaotic Map (MLCSCM)-based multi-server authentication protocol with the key agreement and analyzed its security and performance characteristics. Additionally, we demonstrated the generalization of the proposed MLCSCM for application in multi-server platforms. Compared to the recently published literature, our proposed protocol shows high efficiency and unique security features, provides fierce resistance to numerous attacks, and achieves perfect forward secrecy. However, the detailed security assurance is that our proposed procedure offers only a slight, very fair increase in computation cost due to the use of the MLCSCM and hash operations. Finally, we can conclude that the proposed procedure is both safe and highly efficient. In future work, the potentials of the proposed procedure would be harnessed to provide an efficient dynamic identity-based authentication procedure for multi-gateway wireless sensor networks operating in multi-server environments.

Acknowledgements The authors would like to thank the anonymous reviewers of the Journal of Supercomputing for their excellent reviews and helpful comments and extend their gratitude to the Deanship of Scientific Research at King Khalid University for funding this work through research groups program under grant number R. G. P. 1/72/42. Agbotiname Lucky Imoize is partly supported by the Nigerian Petroleum Technology Development Fund (PTDF) and the German Academic Exchange Service (DAAD) through the Nigerian-German Postgraduate Program under Grant 57473408.

Author contributions CM and RWI conceived the study; ALI contributed to formal analysis; CM, RWI, and SSJ contributed to investigation, methodology, supervision, validation/visualization, and writing—original draft; CM contributed to resources; SGM and ALI contributed to software; CM, RWI, SSJ, SGM, and ALI performed writing—review & editing.

Declarations

Conflict of interest The authors have declared no conflict of interest.

Human and animal rights This article does not contain any studies with human or animal subjects.

References


1. Wang Y, Li C, Khan MA, Li N, Yuan R (2021) Firm information disclosure environment and R&D investment: evidence from Internet penetration. *PLoS ONE* 16(3):1–20. <https://doi.org/10.1371/journal.pone.0247549>
2. Gannon DB, Rosendale V (1984) On the impact of communication complexity on the design of parallel numerical algorithms. *IEEE Trans Comput* 33(12):1180–1194. <https://doi.org/10.1109/TC.1984.1676393>
3. Li CT, Lee CC (2011) A robust remote user authentication scheme using smart card. *Inf Technol Control* 40(3):236–245. <https://doi.org/10.5755/j01.itc.40.3.632>
4. Li CT, Lee CC, Weng CY, Fan CI (2013) An extended multi-server-based user authentication and key agreement scheme with user anonymity. *KSII Trans Internet Inf Syst* 7(1):119–131. <https://doi.org/10.3837/tiis.2013.01.008>
5. Menkus B (1988) Understanding the use of passwords. *Comput Secur* 7(2):132–136. [https://doi.org/10.1016/0167-4048\(88\)90325-2](https://doi.org/10.1016/0167-4048(88)90325-2)
6. Tsaur W-J, Li J-H, Lee W-B (2012) An efficient and secure multi-server authentication scheme with key agreement. *J Syst Softw* 85(4):876–882. <https://doi.org/10.1016/j.jss.2011.10.049>
7. Kohl JT, Neuman BC, Theodore Y (1991) The evolution of the Kerberos authentication service. In: *European Conf. Proc.*, pp 295–313
8. Tsai J-L (2008) Efficient multi-server authentication scheme based on one-way hash function without verification table. *Comput Secur* 27(3):115–121. <https://doi.org/10.1016/j.cose.2008.04.001>
9. Zhu H (2015) A provable one-way authentication key agreement scheme with user anonymity for multi-server environment. *KSII Trans Internet Inf Syst* 9(2):811–828. <https://doi.org/10.3837/tiis.2015.02.019>
10. Lee C-C, Lou D-C, Li C-T, Hsu C-W (2014) An extended chaotic-maps-based protocol with key agreement for multiserver environments. *Nonlinear Dyn* 76(1):853–866. <https://doi.org/10.1007/s11071-013-1174-3>
11. Banerjee S, Dutta MP, Bhunia CT (2015) An improved smart card based anonymous multi-server remote user authentication scheme. *Int J Smart Home* 9(5):11–22. <https://doi.org/10.14257/ijsh.2015.9.5.02>
12. Sun Q, Moon J, Choi Y, Won D (2016) An improved dynamic ID based remote user authentication scheme for multi-server environment. In: Huang X, Xiang Y, Li K-C (eds) *Green, pervasive, and cloud computing*. Springer, Cham, pp 229–242
13. Li X et al (2016) A novel chaotic maps-based user authentication and key agreement protocol for multi-server environments with provable security. *Wirel Pers Commun* 89(2):569–597. <https://doi.org/10.1007/s11277-016-3293-x>
14. Irshad A et al (2018) An enhanced and provably secure chaotic map-based authenticated key agreement in multi-server architecture. *Arab J Sci Eng* 43(2):811–828. <https://doi.org/10.1007/s13369-017-2764-z>
15. Jangirala S, Mukhopadhyay S, Das AK (2017) A multi-server environment with secure and efficient remote user authentication scheme based on dynamic ID using smart cards. *Wirel Pers Commun* 95(3):2735–2767. <https://doi.org/10.1007/s11277-017-3956-2>
16. Ying B, Nayak A (2019) Lightweight remote user authentication protocol for multi-server 5G networks using self-certified public key cryptography. *J Netw Comput Appl* 131:66–74. <https://doi.org/10.1016/j.jnca.2019.01.017>
17. Meshram C, Obaidat MS, Tembhurne JV, Shende SW, Kalare KW, Meshram SG (2020) A lightweight provably secure digital short-signature technique using extended chaotic maps for human-centered IoT systems. *IEEE Syst J*. <https://doi.org/10.1109/JSYST.2020.3043358>
18. Meshram C, Ibrahim RW, Obaid AJ, Meshram SG, Meshram A, El-Latif AMA (2020) Fractional chaotic maps based short signature scheme under human-centered IoT environments. *J Adv Res*. <https://doi.org/10.1016/j.jare.2020.08.015>
19. Meshram C, Lee CC, Meshram SG, Meshram A (2020) OOS-SSS: an efficient online/offline subtree-based short signature scheme using Chebyshev chaotic maps for wireless sensor network. *IEEE Access* 8:80063–80073. <https://doi.org/10.1109/ACCESS.2020.2991348>
20. Meshram C, Li C-T, Meshram SG (2019) An efficient online/offline ID-based short signature procedure using extended chaotic maps. *Soft Comput* 23(3):747–753. <https://doi.org/10.1007/s00500-018-3112-2>

21. Meshram C, Ibrahim RW, Deng L, Shende SW, Meshram SG, Barve SK (2021) A robust smart card and remote user password-based authentication protocol using extended chaotic maps under smart cities environment. *Soft Comput* 25(15):10037–10051. <https://doi.org/10.1007/s00500-021-05929-5>
22. Meshram C, Obaidat MS, Meshram A (2020) An efficient robust lightweight remote user authentication protocol using extended chaotic maps. In: *Proceedings of 2020 International Conference on Computer, Information and Telecommunication Systems CITS 2020*, pp 8–13. <https://doi.org/10.1109/CITS49457.2020.9232622>
23. Datta D et al (2021) An efficient sound and data steganography based secure authentication system. *Comput Mater Contin* 67(1):723–751. <https://doi.org/10.32604/cmc.2021.014802>
24. Meshram C, Lee CC, Ranadive AS, Li CT, Meshram SG, Tembhurne JV (2020) A subtree-based transformation model for cryptosystem using chaotic maps under cloud computing environment for fuzzy user data sharing. *Int J Commun Syst* 33(7):1–15. <https://doi.org/10.1002/dac.4307>
25. Meshram C, Lee C-C, Meshram SG, Li C-T (2019) An efficient ID-based cryptographic transformation model for extended chaotic-map-based cryptosystem. *Soft Comput* 23(16):6937–6946. <https://doi.org/10.1007/s00500-018-3332-5>
26. Meshram C, Ibrahim RW, Obaidat MS, Sadoun B, Meshram SG, Tembhurne JV (2021) An effective mobile-healthcare emerging emergency medical system using conformable chaotic maps. *Soft Comput* 25(14):8905–8920. <https://doi.org/10.1007/s00500-021-05781-7>
27. Gaikwad VP, Tembhurne JV, Meshram C, Lee C-C (2021) Provably secure lightweight client authentication scheme with anonymity for TMIS using chaotic hash function. *J Supercomput.* <https://doi.org/10.1007/s11227-020-03553-y>
28. Kumar P et al (2021) PPSF: a privacy-preserving and secure framework using blockchain-based machine-learning for IoT-driven smart cities. *IEEE Trans Netw Sci Eng.* <https://doi.org/10.1109/TNSE.2021.3089435>
29. Wang W, Xu H, Alazab M, Gadekallu TR, Han Z, Su C (2021) Blockchain-based reliable and efficient certificateless signature for IIoT devices. *IEEE Trans Ind Inform.* <https://doi.org/10.1109/TII.2021.3084753>
30. Shunmuganathan S, Saravanan RD, Palanichamy Y (2015) Secure and efficient smart-card-based remote user authentication scheme for multiserver environment. *Can J Electr Comput Eng* 38(1):20–30. <https://doi.org/10.1109/CJECE.2014.2344447>
31. Mason JC, Handscomb DC (2002) Chebyshev polynomials. CRC Press
32. Bergamo P, D'Arco P, De Santis A, Kocarev L (2005) Security of public-key cryptosystems based on Chebyshev polynomials. *IEEE Trans Circuits Syst I Regul Pap* 52(7):1382–1393. <https://doi.org/10.1109/TCSI.2005.851701>
33. Haubold HJ, Mathai AM, Saxena RK (2011) Mittag–Leffler functions and their applications. *J Appl Math* 2011:298628. <https://doi.org/10.1155/2011/298628>
34. Rahman G, Baleanu D, Al Qurashi M, Purohit SD, Mubeen S, Arshad M (2017) The extended Mittag–Leffler function via fractional calculus. *J Nonlinear Sci Appl* 10(8):4244–4253. <https://doi.org/10.2243/jnsa.010.08.19>
35. Rashid S, Sultana S, Hammouch Z, Jarad F, Hamed YS (2021) Novel aspects of discrete dynamical type inequalities within fractional operators having generalized \hbar -discrete Mittag–Leffler kernels and application. *Chaos Solitons Fractals* 151:111204. <https://doi.org/10.1016/j.chaos.2021.111204>
36. Han S, Chang E (2009) Chaotic map based key agreement with/out clock synchronization. *Chaos Solitons Fractals* 39(3):1283–1289. <https://doi.org/10.1016/j.chaos.2007.06.030>
37. Lee C-C, Hsu C-W (2013) A secure biometric-based remote user authentication with key agreement scheme using extended chaotic maps. *Nonlinear Dyn* 71(1):201–211. <https://doi.org/10.1007/s11071-012-0652-3>
38. Zhang L (2008) Cryptanalysis of the public key encryption based on multiple chaotic systems. *Chaos Solitons Fractals* 37(3):669–674. <https://doi.org/10.1016/j.chaos.2006.09.047>
39. Burrows M, Abadi M, Needham R (1990) A logic of authentication. *ACM Trans Comput Syst* 8(1):18–36. <https://doi.org/10.1145/77648.77649>
40. Wessels J (2001) Applications of Ban-logic. *CMG FINANCE BV* 19:1–23
41. He D, Chen Y, Chen J (2012) Cryptanalysis and improvement of an extended chaotic maps-based key agreement protocol. *Nonlinear Dyn* 69(3):1149–1157. <https://doi.org/10.1007/s11071-012-0335-0>

42. He D, Ma M, Zhang Y, Chen C, Bu J (2011) A strong user authentication scheme with smart cards for wireless communications. *Comput Commun* 34(3):367–374. <https://doi.org/10.1016/j.comcom.2010.02.031>
43. Meshram C, Powar PL (2016) An efficient identity-based QER cryptographic scheme. *Complex Intell Syst* 2(4):285–291. <https://doi.org/10.1007/s40747-016-0030-8>

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Authors and Affiliations

Chandrashekhar Meshram¹  · Rabha W. Ibrahim² · Sarita Gajbhiye Meshram³ · Sajjad Shaukat Jamal⁴ · Agbotiname Lucky Imoize^{5,6}

Rabha W. Ibrahim
rabhaibrahim@yahoo.com

Sarita Gajbhiye Meshram
gajbhiesarita@gmail.com

Sajjad Shaukat Jamal
shussain@kku.edu.sa

Agbotiname Lucky Imoize
aimoize@unilag.edu.ng

- ¹ Department of Post Graduate Studies and Research in Mathematics, Jayawanti Haksar Government Post-Graduation College, College of Chhindwara University, Betul, M.P. 460001, India
- ² IEEE: 94086547, 59200 Kuala Lumpur, Malaysia
- ³ Department for Management of Science and Technology Development, Ton DucThang University, Ho Chi Minh City, Vietnam
- ⁴ Department of Mathematics, College of Science, King Khalid University, Abha, Saudi Arabia
- ⁵ Department of Electrical and Electronics Engineering, Faculty of Engineering, University of Lagos, Akoka 100213, Lagos, Nigeria
- ⁶ Department of Electrical Engineering and Information Technology, Institute of Digital Communication, Ruhr University, 44801 Bochum, Germany

Terms and Conditions

Springer Nature journal content, brought to you courtesy of Springer Nature Customer Service Center GmbH (“Springer Nature”).

Springer Nature supports a reasonable amount of sharing of research papers by authors, subscribers and authorised users (“Users”), for small-scale personal, non-commercial use provided that all copyright, trade and service marks and other proprietary notices are maintained. By accessing, sharing, receiving or otherwise using the Springer Nature journal content you agree to these terms of use (“Terms”). For these purposes, Springer Nature considers academic use (by researchers and students) to be non-commercial.

These Terms are supplementary and will apply in addition to any applicable website terms and conditions, a relevant site licence or a personal subscription. These Terms will prevail over any conflict or ambiguity with regards to the relevant terms, a site licence or a personal subscription (to the extent of the conflict or ambiguity only). For Creative Commons-licensed articles, the terms of the Creative Commons license used will apply.

We collect and use personal data to provide access to the Springer Nature journal content. We may also use these personal data internally within ResearchGate and Springer Nature and as agreed share it, in an anonymised way, for purposes of tracking, analysis and reporting. We will not otherwise disclose your personal data outside the ResearchGate or the Springer Nature group of companies unless we have your permission as detailed in the Privacy Policy.

While Users may use the Springer Nature journal content for small scale, personal non-commercial use, it is important to note that Users may not:

1. use such content for the purpose of providing other users with access on a regular or large scale basis or as a means to circumvent access control;
2. use such content where to do so would be considered a criminal or statutory offence in any jurisdiction, or gives rise to civil liability, or is otherwise unlawful;
3. falsely or misleadingly imply or suggest endorsement, approval, sponsorship, or association unless explicitly agreed to by Springer Nature in writing;
4. use bots or other automated methods to access the content or redirect messages
5. override any security feature or exclusionary protocol; or
6. share the content in order to create substitute for Springer Nature products or services or a systematic database of Springer Nature journal content.

In line with the restriction against commercial use, Springer Nature does not permit the creation of a product or service that creates revenue, royalties, rent or income from our content or its inclusion as part of a paid for service or for other commercial gain. Springer Nature journal content cannot be used for inter-library loans and librarians may not upload Springer Nature journal content on a large scale into their, or any other, institutional repository.

These terms of use are reviewed regularly and may be amended at any time. Springer Nature is not obligated to publish any information or content on this website and may remove it or features or functionality at our sole discretion, at any time with or without notice. Springer Nature may revoke this licence to you at any time and remove access to any copies of the Springer Nature journal content which have been saved.

To the fullest extent permitted by law, Springer Nature makes no warranties, representations or guarantees to Users, either express or implied with respect to the Springer nature journal content and all parties disclaim and waive any implied warranties or warranties imposed by law, including merchantability or fitness for any particular purpose.

Please note that these rights do not automatically extend to content, data or other material published by Springer Nature that may be licensed from third parties.

If you would like to use or distribute our Springer Nature journal content to a wider audience or on a regular basis or in any other manner not expressly permitted by these Terms, please contact Springer Nature at

onlineservice@springernature.com