Computers, Materials & Continua

Tech Science Press

DOI: 10.32604/cmc.2023.037324
*Article*

# An Efficient Three-Party Authenticated Key Exchange Procedure Using Chebyshev Chaotic Maps with Client Anonymity

**Akshaykumar Meshram[1,2], Monia Hadj Alouane-Turki[3], N. M. Wazalwar[2] and Chandrashekhar Meshram[4,*]**

[1]Department of Applied Mathematics, Yeshwantrao Chavan College of Engineering, Nagpur, 441110, Maharashtra, India
[2]Department of Statistics, Rashtrasant Tukadoji Maharaj Nagpur University, Nagpur, 440033, Maharashtra, India
[3]Department of Computer Science, College of Computer Science, King Khalid University, Abha, Saudi Arabia
[4]Department of Post Graduate Studies and Research in Mathematics, Jaywanti Haksar Government Post-Graduate College, College of Chhindwara University, Betul, 460001, Madhya Pradesh, India
*Corresponding Author: Chandrashekhar Meshram. Email: cs_meshram@rediffmail.com
Received: 30 October 2022; Accepted: 22 February 2023

**Abstract:** Internet of Things (IoT) applications can be found in various industry areas, including critical infrastructure and healthcare, and IoT is one of several technological developments. As a result, tens of billions or possibly hundreds of billions of devices will be linked together. These smart devices will be able to gather data, process it, and even come to decisions on their own. Security is the most essential thing in these situations. In IoT infrastructure, authenticated key exchange systems are crucial for preserving client and data privacy and guaranteeing the security of data-in-transit (e.g., via client identification and provision of secure communication). It is still challenging to create secure, authenticated key exchange techniques. The majority of the early authenticated key agreement procedure depended on computationally expensive and resource-intensive pairing, hashing, or modular exponentiation processes. The focus of this paper is to propose an efficient three-party authenticated key exchange procedure (AKEP) using Chebyshev chaotic maps with client anonymity that solves all the problems mentioned above. The proposed three-party AKEP is protected from several attacks. The proposed three-party AKEP can be used in practice for mobile communications and pervasive computing applications, according to statistical experiments and low processing costs. To protect client identification when transferring data over an insecure public network, our three-party AKEP may also offer client anonymity. Finally, the presented procedure offers better security features than the procedures currently available in the literature.

**Keywords:** Client anonymity; Chebyshev chaotic maps; authenticated key exchange; statistical experiment; Galois fields

## 1 Introduction

The term "Internet of Things (IoT)," which is widely described as "a global network of networked devices uniquely reachable, based on standard communication protocols," may have been coined by Kevin Ashton from Procter & Gamble in 1998 [1]. Since then, usage of IoT has sharply increased in part because of the digital revolution (such as widespread connectivity even in developing nations and the ongoing improvement of information and communication technologies). Mobile users may make payments for goods online, pay bills, and conduct other electronic transactions by subscribing to various remote services thanks to the rapid development of low-power and highly efficient networks. Despite being very portable, mobile computing devices are typically insecure and superficial to steal or lose. Without security measures, an unauthorized person might access the data kept on them. For instance, if the data is "sniffed out of the air" during wireless connections or any malware is introduced, intruders may obtain access illegally. Even more severe consequences, such as disabled devices, the loss of personal data, the revelation of non-public data, or charges of misused usage against the device owner, may occur from a lack of authentication and privacy. In addition to the data they hold, mobile computing devices pose a serious security risk since they may give clients access to other services that store or show non-public data. Before distant servers offer clients services for nearly all of these transactions, mutual authentication and client privacy is necessary for the key exchange. Applications for industrial networks [2], wireless sensor networks [3,4], dispersed networks [5–7], as well as Radio Frequency Identification (RFID) systems [8–10] in particular, place a high priority on authentication and privacy.

Many analogous procedures [11–14] have been proposed by researchers to ensure the security of secret keys that are exchanged over an insecure public network, including the Authenticated Key Exchange Procedure (AKEP) based on passwords. The AKEP based on password enables two parties to agree on a standard session key across an insecure public network while maintaining a single memorable password [15–17]. Generally speaking, provided clients use strong passwords that offer adequate entropy, password-based authentication can withstand brute force and dictionary assaults. However, one inherent issue with password-based authentication exists: consumers struggle to memorize long text sequences. Therefore, even though they are aware that the passwords might not be secure, most clients choose memorable passwords, making it difficult to defend against numerous attacks.

Bellovin et al. [18] first introduced the AKEP based on the password procedure in 1992. After ten years, other related procedures have been presented, including the two-party AKEP [11,12] and the three-party AKEP [19,20]. Two-party AKEPs, on the other hand, are not appropriate in the significant peer-to-peer architecture, according to Hassan et al. [21]. Some of the three-party AKEPs are also insufficiently efficient or secure for usage in real-world applications. Recently, in 2005 and 2007, respectively, Abdalla et al. [22], Lu et al. [23], and Algubili et al. [24] introduced three effective three-party AKEPs. Sadly, both methods were still susceptible to offline or online dictionary attacks that were undetectable.

Three-party AKEP based on the password was suggested by Deng et al. [25] in 2009, and it was deemed secure by the universal compostable framework. Deng et al. AKEP, however, is vulnerable to an offline dictionary attack by any other client, according to Yuan et al. [26]. Huang's [27] procedure could not withstand undetectable online dictionary attacks and key-compromise impersonation attacks, according to Yoon et al. [28], who presented a protocol in 2011. Later, Yoon et al. [29] introduced a different procedure and demonstrated how Lou et al. [30] procedure was susceptible to an attacker's off-line password guessing attempts. The security flaws in Huang's [27] procedure were

later discovered by Wu et al. [31], who also suggested a three-party AKEP based on the password to fix the issues with Huang's procedure. The procedure developed by Wu et al. required the most difficult computations due to its numerous exponential computations. Also, their procedure was unable to guarantee client anonymity. Table 1 provides a brief comparison of related research with our proposed AKEP.

**Table 1:** Comparison of our proposed AKEP with related works

| Ref. | Related works | Limitations of related works | Our proposed AKEP |
|---|---|---|---|
| Deng et al. [25] | A three-party password-based key exchange scheme was proposed. | The scheme is diffident contrary to offline dictionary offensive by any other client. | The proposed AKEP is resistant against off-line dictionary attacks. |
| Huang's [27] | A simple three-party password-based key exchange protocol was projected. | The scheme could not prevent subtle online dictionary offensives and key-compromise impersonation offensive. | The proposed AKEP is resistant against detectable online dictionary attacks. |
| Lou et al. [30] | An efficient three-party password-based key exchange scheme was proposed. | The scheme is vulnerable to an offensive's off-line password-guessing insiders or outsiders. | The projected AKEP is resistance against off-line dictionary attacks. |
| Wu et al. [31] | Three-party AKEP, based on the password to fix the issues with Huang's scheme was reported. | It is seen to take the most difficult computations due to its numerous exponential computations. Also, their procedure was unable to guarantee client anonymity. | The proposed work is based on Chebyshev's chaotic maps-based DLP problems that can achieve client anonymity. |
| Zhang et al. [32] | A three-party-AKEP based on the standard model's verification feature was presented. | The model uses anonymous authentication for server and client authentication to strengthen procedure security, which raises computation costs. | Our proposed model provides an extremely low computational processing time. |
| Shu et al. [33] | A three-party-AKEP based on the verification element of the ideal lattice was projected. | The scheme requires six rounds of communication to negotiate a session key, increasing communication overhead. | The proposed work is based on Chebyshev's chaotic maps. The development of the three-party-AKEP scheme depends on the Chebyshev polynomial, which facilitates low communication costs. |

### *1.1 Motivation*

It is challenging to create a three-party AKEP that is both lightweight and secure. Numerous three-party-AKEP have been put forth and afterward shown to be insecure. For instance, Zhang et al. [32] suggested a three-party-AKEP based on the standard model's verification feature in 2019. It employs anonymous authentication for server and client authentication to strengthen procedure security, which raises computation costs. A three-party-AKEP based on the verification element on the ideal lattice was proposed by Shu et al. [33] in 2021. It reduces space complexity but requires six rounds of communication to negotiate a session key, increasing communication overhead. Peikert's [34] error reconciliation mechanism was adopted by Shu et al. [33].

### *1.2 Contribution*

This paper introduced a three-party-AKEP with user anonymity using Chebyshev chaotic maps to increase security and efficiency.

- The presented three-party AKEP encrypts and decrypts the data sent by the user or the server using Chebyshev chaotic maps.
- The presented three-party AKEP can also offer client and server mutual authentication and client anonymity to ensure that clients' identities, which are conveyed via an insecure public network, are guaranteed.
- The presented three-party AKEP has low computing and communication costs and can fend off various attacks, according to the security, statistical experiment and performance analysis.

### *1.3 Paper Organization*

The rest of this paper is structured as follows. The definitions of Chebyshev chaotic maps are introduced briefly in Section 2. Section 3 presents the proposed AKEP. In Section 4, we analyze the projected AKEP and demonstrate that it can withstand several attacks. Section 5 will examine the performance of our AKEP using the former protocols and statistical experiments. Finally, in Section 6, we present our conclusion.

## 2 Background and Material

The section provides a brief overview of the trigonometry in Galois fields, Chebyshev polynomials over Galois fields that are used in the proposed procedure, and the necessary security notations.

### *2.1 Trigonometry in Galois Fields ($\mathcal{GF}$)*

Here we assume that calculations over $\mathcal{GF}(q_1)$, where $q_1 = p^{\iota}$, where $\iota$ is a positive (+) integer and $p$ is an odd prime, are done modulo an irreducible polynomial $f(v)$ of degree whose coefficients are in $\mathcal{GF}(p)$.

***Definition 1.*** The set $\mathcal{GJ}_{(q_1)} = \{s + tr, \forall s, t \in \mathcal{GF}(q_1)\}$ contains the Gaussian integers over $\mathcal{GF}(q_1)$ in a way that $r^2 = -1$ is a quadratic non-residue over $\mathcal{GF}(q_1)$, i.e., $q_1 \equiv 3(mod\ 4)$. Specifically, the extension field $\mathcal{GF}(q_1^2)$ is isomorphic to the "complex" structure $\mathcal{GF}(q_1)$, where each component $\chi = s + tr$ is composed of two parts, $s = \{\chi\}$ (the "actual" part) and $t = \{\chi\}$ (the "imaginary" part).

***Definition 2.*** Let $\chi > 0$ represent an element of $\mathcal{GF}(q_1)$, where $q_1 \equiv 3(mod\,4)$ and has the multiplicative order indicated by ord $(\chi)$. The $\tau-$trigonometric functions identified with $\tau-$cosine and $\tau-$sine are calculated modulo $f(v)$ as

$$cos_\chi(z) := \frac{\chi^z + \chi^{-z}}{2} \text{ and } sin_\chi(z) := \frac{\chi^z - \chi^{-z}}{2r} \tag{1}$$

In this case, we modify the notational scheme established in [35]. Properties like the unit circle and the addition of arcs [35] are naturally shared by the $\tau-$trigonometric functions and the conventional real-valued trigonometric functions. The $\tau$-*cosine* function has period $ord(\chi)$ and even symmetry, as shown in Eq. (1).

$$cos_\chi(z) = cos_\chi(-z(mod\ ord\ (\chi))) \tag{2}$$

The description of the $\tau$-cosine function also depends heavily on the surrounding lemmas. Reference [36] discusses how to prove lemmas, assertions, and theorems.

**Lemma 1.** For $1 \le z, v \le \lfloor ord(\chi) - 1)/2 \rfloor$, where $\lfloor . \rfloor$ denotes the flour function of the argument, $cos_\chi(v) = cos_\chi(z) \Leftrightarrow v = z$.

**Definition 3.** The unimodular set of $\mathcal{GJ}(q_1)$, signified by $\mathcal{G}_1$, is the prearrangement of exponents $\chi = (s + tr,) \in \mathcal{GJ}(q_1)$, such that $s^2 + t^2 \equiv 1(mod\ f(v))$.

**Proposition 1.** $\chi^{q_1+1} \equiv |\chi|^2 \equiv s^2 + t^2 \equiv 1(mod\ f(v))$.

**Proposition 2.** The structure $(\mathcal{G}_1, *)$ is a cyclic group of order $(q_1 + 1)$.

**Lemma 2.** If $\chi = (s + tr,) \in \mathcal{GJ}_{(q_1)}$ is a unimodular exponent, then $cos_\chi(z) = \mathcal{R}\{\chi^z\}, z = 0, 1, \ldots, ord(\chi) - 1$.

**Example 1.** Consider $\mathcal{GF}(27)$. The primitive polynomial $f(v) = v^3 + 2v + 1$ is used to make the Galois field. Let $\chi_1 = (v^2 + 2v) + (v^2 + v + 1)r$ be a unimodular element with order $ord(\chi_1) = 28$ in $\mathcal{GJ}(27)$. Table 2 includes all possible values for $cos_{\chi_1}(z)$. By choosing a specific element $\chi_1 \in \mathcal{GJ}(q_1)$ with multiplicative order ord$\{\mathcal{J}_{\chi_1}\}$, its $\tau-$cosine function may be expressed as $cos_{\chi_1}(z) : \mathsf{Z}_{ord(\chi_1)} \to \mathcal{J}_{\chi_1}$, where $\mathsf{Z}_{ord(\chi_1)}$ is the set of integers modulo $ord(\chi_1)$ and $\mathcal{J}_{\chi_1}$ is the image set of $cos_{\chi_1}$. Note that ord$\{\mathcal{J}_{\chi_1}\} = 15$, which means that the function $arccos_{\chi_1}(z)$ is not defined for each $z \in \mathcal{GF}(27)$.

**Table 2:** All probable estimate for $cos_{\chi_1}(z)$, where $\chi_1 = (v^2 + 2v) + (v^2 + v + 1)r$ is a unimodular component of order 28 in $\mathcal{GJ}(27)$

| $z$ | $cos_{\chi_1}(z)$ | $z$ | $cos_{\chi_1}(z)$ | $z$ | $cos_{\chi_1}(z)$ | $z$ | $cos_{\chi_1}(z)$ |
|---|---|---|---|---|---|---|---|
| 0 | 1 | 7 | 0 | 14 | 2 | 21 | 0 |
| 1 | $v^2 + 2v$ | 8 | $2v^2 + 2v + 2$ | 15 | $2v^2 + v$ | 22 | $v^2 + v + 1$ |
| 2 | $v^2$ | 9 | $v^2 + v$ | 16 | $2v^2$ | 23 | $v^2 + 2v$ |
| 3 | $v^2 + 2$ | 10 | $v^2 + 2v + 1$ | 17 | $2v^2 + 1$ | 24 | $2v^2 + v + 2$ |
| 4 | $2v^2 + v + 2$ | 11 | $2v^2 + 1$ | 18 | $v^2 + 2v + 1$ | 25 | $v^2 + 2$ |
| 5 | $2v^2 + 2v$ | 12 | $2v^2$ | 19 | $v^2 + v$ | 26 | $v^2$ |
| 6 | $v^2 + v + 1$ | 13 | $2v^2 + v$ | 20 | $2v^2 + 2v + 2$ | 27 | $v^2 + 2v$ |

Even if $\chi_1 \in \mathcal{GJ}(q_1)$ has the highest possible multiplicative order $(q_1 + 1)$, as shown in Example 1, the function $arccos_{\chi_1}(z)$ is not described for all $z \in \mathcal{GF}(q_1)$. Thus, we must select a component $\chi_2 (\neq \chi_1)$

yielding $\mathcal{J}_{\chi_1} \cup \mathcal{J}_{\chi_2} = \mathcal{GF}(q_1)$ in order to calculate the inverse $\tau$-*cosine* function of the components that are in $\mathcal{GF}(q_1)$ but not in $\mathcal{J}_{\chi_1}$. The following theorem provides evidence for the existence of such a factor.

**Theorem 1.** Let $\chi_1 \in \mathcal{GJ}(q_1)$ be a unimodular component such that ord $\chi_1 = q_1 + 1$ and $\chi_2 \in \mathcal{GJ}(q_1)$ such that ord $(\chi_2) = q_1 - 1$. Then, $\mathcal{J}_{\chi_1} \cup \mathcal{J}_{\chi_2} = \mathcal{GF}(q_1)$.

**Example 2.** Let $\chi_2 = \mathfrak{v}$ be a component of order 26 in $\mathcal{GF}(27)$, corresponding to the Galois fields produced by the fundamental polynomial $f(\mathfrak{v}) = \mathfrak{v}^3 + 2\mathfrak{v} + 1$. The full range of values for $cos_{\chi_2}(z)$ is displayed in Table 3. Be aware that if we consider the set $\mathcal{J}_{\chi_1}$ (see Example 1), we have $\mathcal{J}_{\chi_1} \cup \mathcal{J}_{\chi_2} = \mathcal{GF}(q_1)$ because $\chi_2$ was chosen by Theorem 1.

**Table 3:** All probable estimates of $cos_{\chi_2}(z)$, where $\chi_2 = \mathfrak{v}$ is a element of order 26 in $\mathcal{GF}(27)$

| $z$ | $cos_{\chi_2}(z)$ | $z$ | $cos_{\chi_2}(z)$ | $z$ | $cos_{\chi_2}(z)$ | $z$ | $cos_{\chi_2}(z)$ |
|---|---|---|---|---|---|---|---|
| 0 | *1* | 7 | $2\mathfrak{v}+2$ | 14 | $2\mathfrak{v}^2+\mathfrak{v}+1$ | 20 | $\mathfrak{v}+1$ |
| 1 | $\mathfrak{v}^2+2\mathfrak{v}+2$ | 8 | $\mathfrak{v}$ | 15 | $2\mathfrak{v}+1$ | 21 | $2\mathfrak{v}$ |
| 2 | $\mathfrak{v}+2$ | 9 | $\mathfrak{v}^2+\mathfrak{v}+2$ | 16 | $\mathfrak{v}^2+2\mathfrak{v}+2$ | 22 | $2\mathfrak{v}^2+2\mathfrak{v}+1$ |
| 3 | $\mathfrak{v}^2+1$ | 10 | $\mathfrak{v}^2+2\mathfrak{v}+2$ | 17 | $\mathfrak{v}^2+\mathfrak{v}+2$ | 23 | $\mathfrak{v}^2+1$ |
| 4 | $2\mathfrak{v}^2+2\mathfrak{v}+1$ | 11 | $2\mathfrak{v}+1$ | 18 | $\mathfrak{v}$ | 24 | $\mathfrak{v}+2$ |
| 5 | $2\mathfrak{v}$ | 12 | $2\mathfrak{v}^2+\mathfrak{v}+1$ | 19 | $2\mathfrak{v}+2$ | 25 | $\mathfrak{v}^2+2\mathfrak{v}+2$ |
| 6 | $\mathfrak{v}+1$ | 13 | 2 | | | | |

### *2.2 Chebyshev Polynomials (CP) over Galois Fields*

In this section, we will look at the properties of CP over Galois fields. There is a definition in [37] that uses the $\tau$-*cosine* function and is in perfect relationship with the CP over $\mathcal{R}$ (real numbers) [38].

**Definition 4.** The CP over $\mathcal{GF}(q_1)$ are defined as

$$\Upsilon_d(z) := cos_\chi(d \times cos_\chi^{-1}(z))(mod\, f(\mathfrak{v})), \tag{3}$$

It can be seen that Eq. (3) is analogous to the arc $\tau$-cosine products. Using the De Moivre, we may generalize a similar situation to real numbers. This yields polynomials of degree $d$ as far as $\tau$-cosines of the respective arcs [38]. However, a simple recurrence relation can be used to obtain Chebyshev chaotic maps for various approximations of $d$. This relationship is determined from Definition 4 for Galois fields via the formula for adding arcs [35], which is

$$\Upsilon_d(z) = 2z\Upsilon_{d-1}(z) - \Upsilon_{d-2}(z)(mod\, f(\mathfrak{v})), \tag{4}$$

where $z \in \mathcal{GF}(q), d \in \mathcal{N}, \Upsilon_0(z) = 1$ and $\Upsilon_1(z) = z$. The following is the periodicity of CP modulo $f(\mathfrak{v})$.

**Proposition 3.** Let $\chi > 0$ be an exponent of $\mathcal{GJ}(q)$ such that $ord(\chi) = \mathcal{N}$. If $z \in \mathcal{J}_\chi$, then $\Upsilon_{a\mathcal{N}\pm d}(z) = \Upsilon_d(z), a \in z$.

Proof: From Definition 4, we have

$$\Upsilon_{a\mathcal{N}\pm d}(z) = cos_\chi\left((a\mathcal{N} \pm d) \times cos_\chi^{-1}(z))\right)(mod\, f(\mathfrak{v}))\Big)$$

After using the formula for the extension of arcs, we can reformulate the preceding statement as

$$\Upsilon_{a\mathcal{N}\pm d}(z) = cos_{\chi}\left(a\mathcal{N} \times cos_{\chi}^{-1}(z)\right) cos_{\chi}\left(d \times cos_{\chi}^{-1}(z)\right)$$

$$\mp sin_{\chi}\left(a\mathcal{N} \times cos_{\chi}^{-1}(z)\right) sin_{\chi}\left(d \times cos_{\chi}^{-1}(z)\right)$$

Since $ord(\chi) = \mathcal{N}$, applying Definition 2, we know that

$cos_{\chi}(a\mathcal{N} \times cos_{\chi}^{-1}(z)) = 1$ and $sin_{\chi}(a\mathcal{N} \times cos_{\chi}^{-1}(z)) = 0$. As a result, we may simplify the last requirement to

$$\Upsilon_{a\mathcal{N}\pm d}(z) = cos_{\chi}\left(d \times cos_{\chi}^{-1}(z)\right) = \Upsilon_{d}(z)$$

If we need to estimate $\Upsilon_d(z)$ for specific values of $z$, and prime power, the restriction that Definition 4 imposes on $z \in \mathcal{J}_{\chi}$ can be eliminated. In this sense, it is possible to use Eq. (4) without relying on the calculation of $\tau$-trigonometric functions.

The semigroup possessions and the chaotic possessions are two significant characteristics that the Chebyshev polynomials display [39,40].

(1) The semigroup possessions:

$$\Upsilon_{\mathfrak{v}}(\Upsilon_{u}(\mathfrak{v})) = \cos\left(\mathfrak{v}\cos^{-1}\left(\cos\left(u\cos^{-1}(\mathfrak{v})\right)\right)\right)$$

$$= \cos\left(\mathfrak{v}u\cos^{-1}(\mathfrak{v})\right) = \Upsilon_{u\mathfrak{v}}(\mathfrak{v})$$

$$= \Upsilon_{u}(\Upsilon_{\mathfrak{v}}(\mathfrak{v}))$$

$\mathfrak{v}$ and $u$ are (+) integer numbers and $\mathfrak{v} \in [-1, 1]$.

(2) The chaotic possessions:

The Chebyshev polynomial map $\Upsilon_{\hbar}(\mathfrak{v}) : [-1, 1] \to [-1, 1]$ of degree $\hbar$ is a chaotic map when the degree $\hbar > 1$ and the (+) Lyapunov exponent $\lambda = \ln \hbar > 0$. Its invariant density is $\mathfrak{f}^{*}(\mathfrak{v}) = 1/(\pi\sqrt{1-\mathfrak{v}^2})$.

It is challenging to solve in polynomial time the following two CP problems [36,38,39]:

(a) The discrete logarithm problem (DLP) is defined as the task of determining the number $\mathfrak{v}$ such that $\Upsilon_{\mathfrak{v}}(\mathfrak{v}) = \mathfrak{w}$ given the two elements $\mathfrak{v}$ and $\mathfrak{w}$.
(b) The Diffie-Hellman problem (DHP) is defined as the task of computing the value $\Upsilon_{\mathfrak{v}u}(\mathfrak{v})$ given three elements $\mathfrak{v}$, $\Upsilon_{\mathfrak{v}}(\mathfrak{v})$, and $\Upsilon_{u}(\mathfrak{v})$.

## 3 The AKEP Based on Chebyshev Chaotic Maps

The projected procedure with user anonymity using extended chaotic maps is detailed in depth in this section. Table 4 summarizes the notations utilized in our protocol.

At first, the distant server $\mathcal{JS}$ chooses a random numbers $u \in_R \mathcal{GF}(q_1)$ and $\mathfrak{v} \in_R \mathcal{GF}(q_1)$, then calculates its public key $\mathcal{P} \equiv \Upsilon_{u}(\mathfrak{v})\,(mod\, f(\mathfrak{v}))$. The remote server $\mathcal{JS}$ conceals its private key $u$. In our protocol, we assume that the clients, $\mathcal{C}_1$ and $\mathcal{C}_2$, have previously established the remote server $\mathcal{JS}$'s shared secret key-sharing passwords $\mathfrak{pwd}_{\mathcal{C}_1}$ and $\mathfrak{pwd}_{\mathcal{C}_2}$. The remote server $\mathcal{JS}$ distributes the public constraints $(\mathcal{P}, \mathfrak{v}, \hbar(\cdot), f(\mathfrak{v}))$ to all network participants. Fig. 1 depicts a simplified depiction

of the suggested protocol. The following steps summarize the details of the proposed protocol from this point forward:

**Table 4:** The symbols used in our presented procedure

| Symbolization | Description |
|---|---|
| $\mathcal{C}_1, \mathcal{C}_2$ | Identity of two different clients. |
| $\mathcal{JS}$ | A $\mathcal{JS}$ (remote server) |
| $pwd_{\mathcal{C}_1}, pwd_{\mathcal{C}_2}$ | The password shared among client $\mathcal{C}_1$(resp.$\mathcal{C}_2$) and server $\mathcal{JS}$ |
| $q_1$ | A prime number |
| $v, u$ | Random numbers selected by $\mathcal{JS}$ |
| $\mathcal{P}$ | The public key of $\mathcal{JS}$, where $\mathcal{P} \equiv \Upsilon_u(\mathfrak{v}) (\mathrm{mod} f(\mathfrak{v}))$, |
| $SK$ | The session key used between clients $\mathcal{C}_1$ and $\mathcal{C}_2$ |
| $\mathfrak{v}, w$ | Randomly selected integers |
| $t_1$ | The time-stamp |
| $\hbar(\cdot)$ | A secure hash function |
| $\|$ | The concatenation operation |
| $\bigoplus$ | The exclusive-or (XOR) operation |

(1) Client $\mathcal{C}_1$ selects an integer $\mathfrak{v}$ at random and calculates the following:

$\mathfrak{R}_{\mathcal{C}_1} \equiv \Upsilon_{\mathfrak{v}}(v) (\mathrm{mod} f(\mathfrak{v}))$, $\Upsilon_{\mathcal{C}_1} \equiv \Upsilon_{\mathfrak{v}}(\mathcal{P}) (\mathrm{mod} f(\mathfrak{v}))$, $\mathcal{C}_1 ID_i = \mathcal{C}_1 \bigoplus \hbar(\Upsilon_{\mathcal{C}_1})$,

$\Upsilon_{\mathcal{C}_1,\mathcal{S}} = \hbar(\mathcal{C}_1 \| \mathcal{C}_2 \| \mathcal{JS} \| \mathcal{C}_1 ID_i \| pwd_{\mathcal{C}_1} \| \Upsilon_{\mathcal{C}_1} \| t_1)$.

Client $\mathcal{C}_1$ sends $(\mathcal{C}_1 ID_i, \mathfrak{R}_{\mathcal{C}_1}, \Upsilon_{\mathcal{C}_1,\mathcal{S}}, t_1)$ to client $\mathcal{C}_2$.

(2) The client then sends back the response $(\mathcal{C}_1 ID_i, \mathfrak{R}_{\mathcal{C}_1}, \Upsilon_{\mathcal{C}_1,\mathcal{S}}, t_1)$.

$\mathcal{C}_2$ selects an integer at random. $w$ and calculates the following

$\mathfrak{R}_{\mathcal{C}_2} \equiv \Upsilon_w(v) (mod f(\mathfrak{v}))$, $\Upsilon_{\mathcal{C}_2} \equiv \Upsilon_w(\mathcal{P}) (mod f(\mathfrak{v}))$ , $\mathcal{C}_2 ID_i = \mathcal{C}_2 \bigoplus \hbar(\Upsilon_{\mathcal{C}_2})$,

$\Upsilon_{\mathcal{C}_2,\mathcal{S}} = \hbar(\mathcal{C}_2 \| \mathcal{JS} \| \mathcal{C}_2 ID_i \| pwd_{\mathcal{C}_2} \| \Upsilon_{\mathcal{C}_2})$.

The second client, $\mathcal{C}_2$, then transmits the remote server, $\mathcal{JS}$, the following data: $\left(\mathcal{C}_1 ID_i, \mathfrak{R}_{\mathcal{C}_1}, \Upsilon_{\mathcal{C}_1,\mathcal{S}}, \mathcal{C}_2 ID_i, \mathfrak{R}_{\mathcal{C}_2}, \Upsilon_{\mathcal{C}_2,\mathcal{S}}, t_1\right)$.

(3) When receiving $(\mathcal{C}_1 ID_i, \mathfrak{R}_{\mathcal{C}_1}, \Upsilon_{\mathcal{C}_1,\mathcal{S}}, \mathcal{C}_2 ID_i, \mathfrak{R}_{\mathcal{C}_2}, \Upsilon_{\mathcal{C}_2,\mathcal{S}}, t_1)$, the server $\mathcal{JS}$ first verifies the accuracy of $t_1$ by determining whether the equation $t' - t_1 > \Delta t$ holds, where $t'$ is the time at which the server gets the messages from $\mathcal{C}_2$. $\Delta t$ stands for the predefined. In the event that the equation is incorrect, the server $\mathcal{JS}$ computes $\Upsilon'_{\mathcal{C}_1} \equiv \Upsilon_u\left(\mathfrak{R}_{\mathcal{C}_1}\right) (\mathrm{mod} f(\mathfrak{v}))$, $\Upsilon'_{\mathcal{C}_2} \equiv \Upsilon_{\mathcal{S}}\left(\mathfrak{R}_{\mathcal{C}_2}\right) (\mathrm{mod} f(\mathfrak{v}))$, $\mathcal{C}'_1 = \mathcal{C}_1 ID_i \bigoplus \hbar(\Upsilon'_{\mathcal{C}_1})$, and $\mathcal{C}'_2 = \mathcal{C}_2 ID_i \bigoplus \hbar(\Upsilon'_{\mathcal{C}_2})$ and uses them to check $\Upsilon_{\mathcal{C}_1,\mathcal{S}}$ and $\Upsilon_{\mathcal{C}_2,\mathcal{S}}$ respectively. The protocol is ended by $\mathcal{JS}$ if the values are incorrect. If not, $\mathcal{JS}$ calculates $\Upsilon_{\mathcal{S},\mathcal{C}_1} = \hbar(\mathcal{C}'_1 \| \mathcal{C}'_2 \| \Upsilon \mathcal{S} \| pwd_{\mathcal{C}_1} \| \Upsilon'_{\mathcal{C}_1})$, $\Upsilon_{\mathcal{S},\mathcal{C}_2} = \hbar(\mathcal{C}'_1 \| \mathcal{C}'_2 \| \mathcal{JS} \| pwd_{\mathcal{C}_2} \| \Upsilon'_{\mathcal{C}_2})$, and $\mathcal{C}_1 ID_j = \mathcal{C}'_1 \bigoplus \hbar(\Upsilon'_{\mathcal{C}_2})$, and then sends $(\Upsilon_{\mathcal{S},\mathcal{C}_1}, \Upsilon_{\mathcal{S},\mathcal{C}_2}, \mathcal{C}_1 ID_j)$ to client $\mathcal{C}_2$.

(4) After receiving $(\Upsilon_{\mathcal{S},\mathcal{C}_1}, \Upsilon_{\mathcal{S},\mathcal{C}_2}, \mathcal{C}_1 ID_j)$, client $\mathcal{C}_2$ computes $\mathcal{C}''_1 = \mathcal{C}_1 ID_j \bigoplus \hbar(\Upsilon_{\mathcal{C}_2})$ and uses $\Upsilon_{\mathcal{C}_2}$ to validate $\Upsilon_{\mathcal{S},\mathcal{C}_2}$. Client $\mathcal{C}_2$ stops the protocol if the value is invalid. Else, both server $\mathcal{JS}$ and client $\mathcal{C}_2$ are authenticated, and client $\mathcal{C}_2$ calculates the shared session key $SK \equiv \Upsilon_w\left(\mathfrak{R}_{\mathcal{C}_1}\right) (\mathrm{mod} f(\mathfrak{v}))$ and $\mathcal{S}_{\mathcal{C}_1\mathcal{C}_2} = \hbar(SK \| \mathcal{C}''_1 \| \mathcal{C}_2)$. Lastly, $\mathcal{C}_2$ sends $(\mathfrak{R}_{\mathcal{C}_2}, \Upsilon_{\mathcal{S},\mathcal{C}_1}, \mathcal{S}_{\mathcal{C}_1\mathcal{C}_2})$ to client $\mathcal{C}_1$.

(5) Client $\mathcal{C}_1$ initially verifies the legitimacy of $\Upsilon_{s,\mathcal{C}_1}$ using $\Upsilon_{\mathcal{C}_1}$ after obtaining $(\Re_{\mathcal{C}_2}, \Upsilon_{s,\mathcal{C}_1}, s_{\mathcal{C}_1\mathcal{C}_2})$. The protocol is ended by $\mathcal{C}_1$ if the value is invalid. If not, client $\mathcal{C}_1$ calculates the shared session key $SK \equiv \Upsilon_{\mathbb{v}}\left(\Re_{\mathcal{C}_2}\right) (\text{mod } f(\mathbb{v}))$ and verifies that $s_{\mathcal{C}_1\mathcal{C}_2} = \hbar(SK \parallel \mathcal{C}_1 \parallel \mathcal{C}_2)$ is valid. $\mathcal{C}_1$ ends the protocol if it doesn't hold. Otherwise, the shared session key $SK$ is agreed upon, and the server $\mathcal{JS}$ and user $\mathcal{C}_1$ are both authenticated. Client $\mathcal{C}_1$ and $\mathcal{C}_2$ can communicate securely using the shared session key $SK$. One session is all that the shared session key $SK$ is used for.

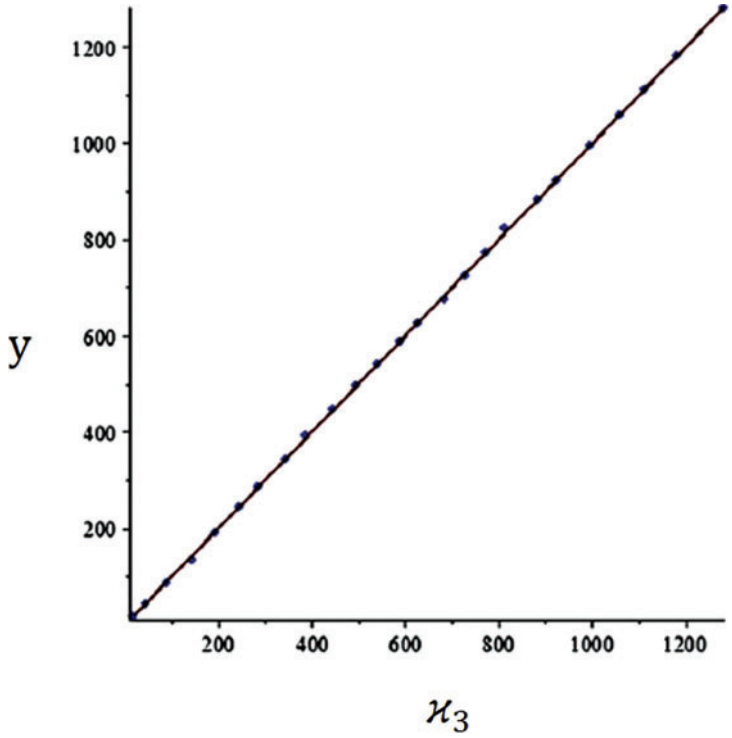
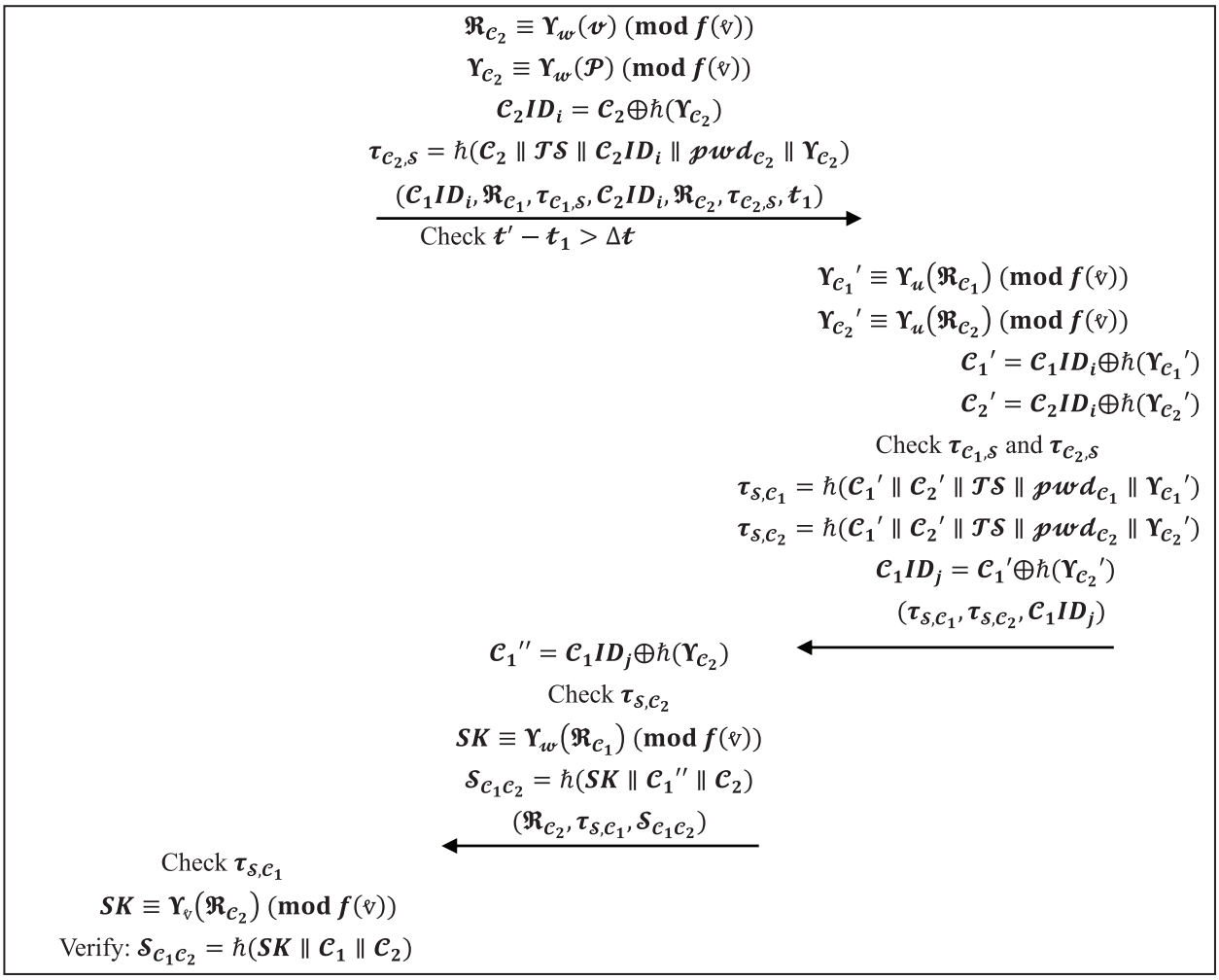



**Figure 1:** Chebyshev-based Decisional Diffie-Hellman problem

| Client $\boldsymbol{\mathcal{C}_1}$ | Client $\boldsymbol{\mathcal{C}_2}$ | Trusted server $\boldsymbol{\mathcal{TS}}$ |
|---|---|---|
| $\boldsymbol{pwd_{\mathcal{C}_1}}$ | $\boldsymbol{pwd_{\mathcal{C}_2}}$ | $(\boldsymbol{u}, \boldsymbol{\mathcal{P}} \equiv \boldsymbol{\Upsilon_u(v)}\ (\textbf{mod}\ \boldsymbol{f}(\mathbb{v})))$ |
| Generate $\mathbb{v}$ | | |
| $\boldsymbol{\Re_{\mathcal{C}_1}} \equiv \boldsymbol{\Upsilon_{\mathbb{v}}(v)}\ (\textbf{mod}\ \boldsymbol{f}(\mathbb{v}))$ | | |
| $\boldsymbol{\Upsilon_{\mathcal{C}_1}} \equiv \boldsymbol{\Upsilon_{\mathbb{v}}(\mathcal{P})}\ (\textbf{mod}\ \boldsymbol{f}(\mathbb{v}))$ | | |
| $\boldsymbol{\mathcal{C}_1 ID_i} = \boldsymbol{\mathcal{C}_1} \oplus \hbar(\boldsymbol{\Upsilon_{\mathcal{C}_1}})$ | | |
| $\boldsymbol{\tau_{\mathcal{C}_1,\mathcal{S}}} = \hbar(\boldsymbol{\mathcal{C}_1} \parallel \boldsymbol{\mathcal{C}_2} \parallel \boldsymbol{\mathcal{TS}} \parallel \boldsymbol{\mathcal{C}_1 ID_i} \parallel \boldsymbol{pwd_{\mathcal{C}_1}} \parallel \boldsymbol{\Upsilon_{\mathcal{C}_1}} \parallel \boldsymbol{t_1})$ | | |
| $(\boldsymbol{\mathcal{C}_1 ID_i}, \boldsymbol{\Re_{\mathcal{C}_1}}, \boldsymbol{\tau_{\mathcal{C}_1,\mathcal{S}}}, \boldsymbol{t_1})$ → | | |
| | Generate $\boldsymbol{w}$ | |




## 4 Security Investigation and Discussion

In this segment, we examine the security and performance of our presented procedure and demonstrate its ability to withstand various attacks. Here, we explain several security analyses in our proposed process.

**Proposition 4.1:** The presented AKEP can accomplish off-line dictionary attacks.

Proof: The attacker might deduce the password from the element $\Upsilon_{\mathcal{C}_1,\mathcal{S}}$ or $\Upsilon_{\mathcal{C}_2,\mathcal{S}}$ in the messages $(\mathcal{C}_1 ID_i, \mathfrak{R}_{\mathcal{C}_1}, \Upsilon_{\mathcal{C}_1,\mathcal{S}}, \mathcal{t}_1)$ or $(\mathcal{C}_1 ID_i, \mathfrak{R}_{\mathcal{C}_1}, \Upsilon_{\mathcal{C}_1,\mathcal{S}}, \mathcal{C}_2 ID_i, \mathfrak{R}_{\mathcal{C}_2}, \Upsilon_{\mathcal{C}_2,\mathcal{S}}, \mathcal{t}_1)$. Client $\mathcal{C}_1$ and $\mathcal{C}_2$ generate $\Upsilon_{\mathcal{C}_1}$ or $\Upsilon_{\mathcal{C}_2}$, respectively, according to the difficulty of the chaotic map based DLP issue, and without knowing either of these, the attacker cannot validate the password. Because of this, our procedure is secure from dictionary attacks in the background.

**Proposition 4.2:** The presented AKEP can accomplish undetectable online dictionary attacks.

Proof: The attacker may attempt to impersonate a legitimate client by intercepting the messages $(\mathcal{C}_1 ID_i, \mathfrak{R}_{\mathcal{C}_1}, \Upsilon_{\mathcal{C}_1,\mathcal{S}}, \mathcal{t}_1)$ or $(\mathcal{C}_1 ID_i, \mathfrak{R}_{\mathcal{C}_1}, \Upsilon_{\mathcal{C}_1,\mathcal{S}}, \mathcal{C}_2 ID_i, \mathfrak{R}_{\mathcal{C}_2}, \Upsilon_{\mathcal{C}_2,\mathcal{S}}, \mathcal{t}_1)$. However, unless the attacker has successfully guessed the correct password, they are unable to send a new valid message

$(\mathcal{C}_1ID_i, \Re_{\mathcal{C}_1}, \Upsilon_{\mathcal{C}_1,\mathcal{S}}, \mathcal{C}_2ID_i, \Re_{\mathcal{C}_2}, \Upsilon_{\mathcal{C}_2,\mathcal{S}}, t_1)$ to the trustworthy server. Additionally, the attacker will run into the the chaotic map-based DLP issue if they attempt to guess the password. As a result, our presented procedure can withstand attacks using an undetected online dictionary.

**Proposition 4.3:** The presented AKEP can accomplish detectable online dictionary attacks.

Proof: The attacker may attempt to impersonate a legitimate client by intercepting the messages $(\mathcal{C}_1ID_i, \Re_{\mathcal{C}_1}, \Upsilon_{\mathcal{C}_1,\mathcal{S}}, t_1)$ or $(\mathcal{C}_1ID_i, \Re_{\mathcal{C}_1}, \Upsilon_{\mathcal{C}_1,\mathcal{S}}, \mathcal{C}_2ID_i, \Re_{\mathcal{C}_2}, \Upsilon_{\mathcal{C}_2,\mathcal{S}}, t_1)$. However, unless the attacker has successfully guessed the correct password, they are unable to send a new valid message $(\mathcal{C}_1ID_i, \Re_{\mathcal{C}_1}, \Upsilon_{\mathcal{C}_1,\mathcal{S}}, \mathcal{C}_2ID_i, \Re_{\mathcal{C}_2}, \Upsilon_{\mathcal{C}_2,\mathcal{S}}, t_1)$ to the trustworthy server. Additionally, the server will verify that $\Upsilon_{\mathcal{C}_1,\mathcal{S}}$ and $\Upsilon_{\mathcal{C}_2,\mathcal{S}}$ are valid. Therefore, if an attacker sends an invalid message to the server, the attacker will be found. In that instance, our presented procedure is secure against detectable online dictionary attacks.

**Proposition 4.4:** The presented AKEP can counterattack replay and impersonation attacks.

Proof: A client's messages may be intercepted by the attacker, who may then replay them to the server in the subsequent run. However, by verifying the accuracy of the timestamp $t_1$, the server was able to identify the attack. The attacker may potentially eavesdrop on server messages and repeat them to the client. However, the new random integers $v$ and $w$ were produced by the clients. The attack might then be discovered by clients $\mathcal{C}_1$ and $\mathcal{C}_2$, respectively, by confirming the accuracy of $\Upsilon_{\mathcal{S},\mathcal{C}_1}$ and $\Upsilon_{\mathcal{S},\mathcal{C}_2}$. Any changes to the preceding parameters during the authentication phase result in discrepancies between the supplied parameters and the received hash value, and the authentication request fails. An attacker cannot replay or change the server authentication message using the same reasoning described previously. As a result, our presented procedure is resistant to both replay and user impersonation attacks.

**Proposition 4.5:** The presented AKEP can achieve user anonymity.

Proof: The attacker may listen in on the client's conversation with the trusted server and attempt to identify the client's true identity to gather some of the client's security-sensitive data. The real identities of clients $\mathcal{C}_1$ and $\mathcal{C}_2$ are shielded in our proposed procedure by $\mathcal{C}_1ID_i = \mathcal{C}_1 \bigoplus \hbar(\Upsilon_{\mathcal{C}_1})$, and $\mathcal{C}_2ID_i = \mathcal{C}_2 \bigoplus \hbar(\Upsilon_{\mathcal{C}_2})$, respectively. The attacker will have to calculate $\Upsilon_{\mathcal{C}_1}$ and $\Upsilon_{\mathcal{C}_2}$ and deal with the chaotic map-based DLP problem. As a result, our presented procedure can offer the user a high level of anonymity.

**Proposition 4.6:** The presented AKEP can accomplish mutual authentication.

Proof: Mutual authentication between the client and the server is possible with our presented procedure. Step 3 of our procedure requires the server $\mathcal{JS}$ to validate $\Upsilon_{\mathcal{C}_1,\mathcal{S}}$ and $\Upsilon_{\mathcal{C}_2,\mathcal{S}}$ to authenticate clients $\mathcal{C}_1$ and $\mathcal{C}_2$. To authenticate server $\mathcal{JS}$, clients $\mathcal{C}_1$ and $\mathcal{C}_2$ must also verify the validity of $\Upsilon_{\mathcal{S},\mathcal{C}_1}$ and $\Upsilon_{\mathcal{S},\mathcal{C}_2}$. If an attacker attempts to forge messages, they will encounter both chaotic map-based DLP and chaotic map-based DHP issues. As a result, because both the client and the trusted server can authenticate each other, mutual authentication is achieved. As a result, mutual authentication is achieved because both the client and the trusted server can authenticate each other.

**Proposition 4.7:** The presented AKEP can prevent denial-of-service (DoS) attacks.

Proof: When $\mathcal{C}_2$ receives the message $\left(\mathcal{C}_1ID_i, \Re_{\mathcal{C}_1}, \tau_{\mathcal{C}_1,\mathcal{S}}, t_1\right)$ from $\mathcal{C}_1$, $\mathcal{C}_2$ first verifies the timestamp's accuracy. If it is invalid, $\mathcal{C}_2$ throws the message away. If not, it calculates a value to compare with the received value: $(\mathcal{C}_1ID_i, \Re_{\mathcal{C}_1}, \Upsilon_{\mathcal{C}_1,\mathcal{S}}, \mathcal{C}_2ID_i, \Re_{\mathcal{C}_2}, \Upsilon_{\mathcal{C}_2,\mathcal{S}}, t_1)$. DoS occurs when an attacker uses the same or different identities to access a specific resource in a hugely and simultaneously. Because the system can preserve the session, access by the same identity to the same resource can be restricted to only one session at a time. The genuine identities of clients $\mathcal{C}_1$ and $\mathcal{C}_2$ are concealed in our suggested process

by $\mathcal{C}_1 ID_i = \mathcal{C}_1 \bigoplus \hbar(\Upsilon_{\mathcal{C}_1})$ and $\mathcal{C}_2 ID_i = \mathcal{C}_2 \bigoplus \hbar(\Upsilon_{\mathcal{C}_2})$, respectively. However, the presented procedure can mitigate DoS attacks to some extent. As described in Section 3(3), the proposed procedure exploits the advantages of timestamp. The proposed procedure can resist such DoS attacks.

**Proposition 4.8:** The presented AKEP can accomplish Bergamo et al.'s attack [41].

Proof: This attack is conceivable in two scenarios. First, an attacker must have the relevant parameters $\mathscr{v}$, $\Upsilon_{\mathscr{v}}(\mathscr{v})$ and $\Upsilon_{\mathscr{w}}(\mathscr{v})$; second, if numerous Chebyshev polynomials reach the same crossing point, the adversary will be able to recover the encrypted text due to the periodicity of cosine functions. The suggested protocol substitutes $\Upsilon_{\mathscr{v}}(\mathscr{v})$ and $\Upsilon_{\mathscr{w}}(\mathscr{v})$ inside $\mathcal{C}_1 ID_i = \mathcal{C}_1 \bigoplus \hbar(\Upsilon_{\mathcal{C}_1})$ and $\mathcal{C}_2 ID_i = \mathcal{C}_2 \bigoplus \hbar(\Upsilon_{\mathcal{C}_2})$, where $\Upsilon_{\mathcal{C}_1} \equiv \Upsilon_{\mathscr{v}}(\mathcal{P}) \pmod{f(\mathscr{v})}$ and $\Upsilon_{\mathcal{C}_2} \equiv \Upsilon_{\mathscr{w}}(\mathcal{P}) \pmod{f(\mathscr{v})}$ correspondingly. Adversaries cannot obtain these polynomials unless they know the hash values of $\Upsilon_{\mathcal{C}_1}$ and $\Upsilon_{\mathcal{C}_2}$, which are transmitted to the client over a secure channel. Furthermore, the protocol's use of augmented Chebyshev polynomials over Galois fields makes Bergamo et al.'s attack unachievable.

## 5 Analysis

### *5.1 Statistical Experiments*

Statistical experiments will be used to demonstrate the Chebyshev-based Decisional Diffie-Hellman (CDDH) assumption. A fast graphical tool for comparing two probability distributions in statistics is the quantile-quantile plot ($Q-Q$ *plot*) [42]. The sample quintiles and theoretical quintiles are contrasted visually. The experimental result will be roughly on the reference line "$\mathcal{Y} = \mathcal{X}$," where $\mathcal{X}$ represents the first data sample, and $\mathcal{Y}$ represents the second data sample, if two data samples are taken from the same distribution. As a result, the $Q-Q$ plot, a non-parametric approach, is widely used in cryptosystem security analysis to demonstrate the distribution indistinguishability among privacy data and uniformly random data. This technique has previously been used to demonstrate the security of proposed procedures [43]. Small size parameters are sufficient in the experiment for the distribution of given arrays. Let $\left(\varkappa_0 = \mathcal{P}, \varkappa_1 = \Upsilon_{\mathcal{C}_1} \equiv \Upsilon_{\mathscr{v}}(\mathcal{P}), \varkappa_2 = \Upsilon_{\mathcal{C}_2} \equiv \Upsilon_{\mathscr{w}}(\mathcal{P}), \varkappa_3 = SK \equiv \Upsilon_{\mathscr{w}}(\Upsilon_{\mathscr{v}}(\mathscr{v})) = \Upsilon_{\mathscr{v}}(\Upsilon_{\mathscr{w}}(\mathscr{v}))\right)$ be a CD DH tuple. The goal is to determine whether $\varkappa_3 \equiv \Upsilon_{\mathscr{v}}(\Upsilon_{\mathscr{w}}(\mathscr{v})) = \Upsilon_{\mathscr{v}\mathscr{w}}(\mathscr{v}) = \Upsilon_{\ell}(\mathscr{v})$, where $\ell = \mathscr{v}\mathscr{w}$ or not. Meanwhile, if $\mathcal{C} = f(\mathscr{v}) = 1319, \varkappa_0 = 2, \mathscr{v} = 460,$ and $\mathscr{w} = 981$, then $\varkappa_1 = 861$ and $\varkappa_2 = 1182$, with $\mathscr{v}$ and $\mathscr{w}$ unknown to the challenger and adversary. Let $\mathrm{y} = \Upsilon_{\mathrm{t}}(\varkappa_i) \pmod{c}$ be a Chebyshev map for $i = 1, 2$. We consider the case where $i = 1$ without losing generality. Now, by sampling 1500 times, we utilize the Q–Q plot to display the distributions of y and $\varkappa_3$ before tallying the frequency with which each $Z_c$ component appears. In the event, that the two distributions are identical, the resulting graph will be a straight line.

"$\mathcal{Y} = \mathcal{X}$"

The vertical axis shows estimated quintiles from the y sample set and the horizontal axis shows estimated quintiles from the $\varkappa_3$ sample set. Using Maple-17, the statistical experiment is run.

Fig. 1 depicts a comparison of the distributions of y and $\varkappa_3$. Because the graph is a straight line, $\Upsilon_{\mathrm{t}_1}(\varkappa_1)$ is statistically equivalent to $\Upsilon_{\mathrm{t}_3}(\varkappa_0)$. Although the $Q-Q$ plot cannot provide theoretical proof of the security, it can provide evidence when the theoretical proof is challenging to achieve.

In addition, we provide the Chebyshev polynomial's function image for random $\ell$: $\mathrm{y} = \Upsilon_{\mathrm{t}_\ell}(\varkappa_0) \pmod{f(\mathscr{v})}$. The period of $\Upsilon_{\ell}(\varkappa_0) \pmod{f(\mathscr{v})}$ is $\frac{c-1}{2}$, as seen in Fig. 2. In reality, it has been demonstrated that the period of $\Upsilon_d(\varkappa) \pmod{c}$ for the odd prime is a divisor of $c-1$ when the

roots of $\xi^2 - 2\varkappa_0\xi + 1 = 0$ are in $GF_c$ [43] and the period of $\Upsilon_d(\varkappa) \pmod c$ is a divisor of $c-1$ when the roots are in $\mathrm{GF}_{c^2}$. Therefore, we choose a prime $c = 2c_0 + 1$, where $c_0$ is likewise a prime, to prevent a short period of $\Upsilon_\ell(\varkappa_0) \pmod c$. Furthermore, the procedure's randomness is based on the pseudo-randomness of $\Upsilon_\ell(\varkappa_0) \pmod c$.

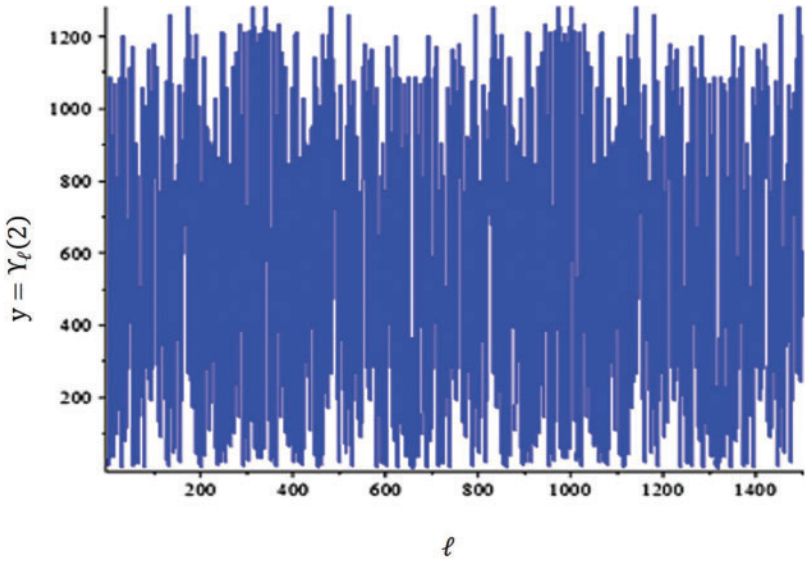



**Figure 2:** Chebyshev polynomials image map

### *5.2 Efficiency and Performance Analysis*

To illustrate the security performance and effectiveness of our new design, we will compare the security properties of our given AKEP with four previous AKEPs introduced by Wu et al. [31], Yanrong et al. [44], Guo et al. [45], and Lee [46] in Table 5. In Table 5, we can see that our AKEP is more secure than other AKEPs. To express the execution time necessary for modular multiplication, one-way hash function, chaotic map operation, group modular exponentiation, and symmetric encryption/decryption operation, notations such as $\dagger_m, \dagger_h, \dagger_c, \dagger_e$, and $\dagger_s$, are used to show our evaluation results. Table 6 shows the client as $\mathcal{C}$ and the server as $\mathcal{JS}$. Table 6 and Fig. 3 shows how our novel AKEP compares to similar procedures such as Wu et al. [31], Yanrong et al. [44], Guo et al. [45], and Lee [46] in terms of communication costs. We arrive at the subsequent communication time figures with unit hashing time based on the experimental findings in [47–51]: $\dagger_e = 600\dagger_h$, $\dagger_m = 2.5\dagger_h$, $\dagger_s = \dagger_h$ and $\dagger_h = \dagger_c$. We obtain the subsequent order of computational complexity using this above relation: $\dagger_h \approx \dagger_c \approx \dagger_s < \dagger_m < \dagger_e$. By the way, we know that the running time of $\dagger_h$ is 0.503 ms [51]. The total communication costs of the work of Wu et al. [31], the work of Yanrong et al. [44], the work of Guo et al. [45], the work of Lee [46], and the projected protocol are 3022.03, 13.59, 11.07, 10.06, and 8.56 ms, respectively. According to the study findings in Fig. 3, the presented AKEP has by far the lowest interaction value. In terms of runtime, the proposed AKEP frequently produces tests that outperform the other procedures.

**Table 5:** Comparison of security properties

| Security properties (SP) | [31] | [44] | [45] | [46] | Proposed |
|---|---|---|---|---|---|
| $SP_1$ | ÿ | Ñ | Ñ | Ñ | ÿ |
| $SP_2$ | ÿ | Ñ | Ñ | Ñ | ÿ |
| $SP_3$ | ÿ | Ñ | Ñ | Ñ | ÿ |
| $SP_4$ | ÿ | ÿ | ÿ | ÿ | ÿ |
| $SP_5$ | Ñ | ÿ | Ñ | ÿ | ÿ |
| $SP_6$ | Ñ | ÿ | Ñ | ÿ | ÿ |
| $SP_7$ | ÿ | ÿ | ÿ | ÿ | ÿ |
| $SP_8$ | Ñ | Ñ | Ñ | Ñ | ÿ |
| $SP_9$ | Ñ | ÿ | Ñ | ÿ | ÿ |

Note: $SP_1$: off-line dictionary attacks; $SP_2$: undetectable online dictionary attacks; $SP_3$: detectable online dictionary attacks; $SP_4$: replay attack; $SP_5$: impersonation attack; $SP_6$: user anonymity; $SP_7$: mutual authentication; $SP_8$: denial-of-service attack; $SP_9$: Bergamo et al.'s attack. ÿ: Secure; Ñ: Vulnerable.

**Table 6:** Computation cost examination of the projected AKEP with other similar procedures

| | [31] | [44] | [45] | [46] | Proposed |
|---|---|---|---|---|---|
| $\mathcal{C}$ (Client) | $8\dagger_e + 6\dagger_h$ | $7\dagger_h + 3\dagger_c + 3\dagger_s$ | $8\dagger_h + 2\dagger_c$ | $6\dagger_h + 2\dagger_c$ | $5\dagger_h + 5\dagger_c$ |
| $\mathcal{JS}$ (Server) | $2\dagger_e + 2\dagger_h$ | $8\dagger_h + 2\dagger_c + 4\dagger_s$ | $10\dagger_h + 2\dagger_c$ | $10\dagger_h + 2\dagger_c$ | $5\dagger_h + 2\dagger_c$ |
| Total communication cost (ms) | $10\dagger_e + 8\dagger_h \approx$ 3022.03 *ms* | $15\dagger_h + 5\dagger_c +$ $7\dagger_s \approx 13.59$ *ms* | $18\dagger_h + 4\dagger_c \approx$ 11.07 *ms* | $16\dagger_h + 4\dagger_c \approx$ 10.06 *ms* | $10\dagger_h + 7\dagger_c \approx$ 8.56 *ms* |
| No. of message communications | 5 | 4 | 6 | 3 | 4 |

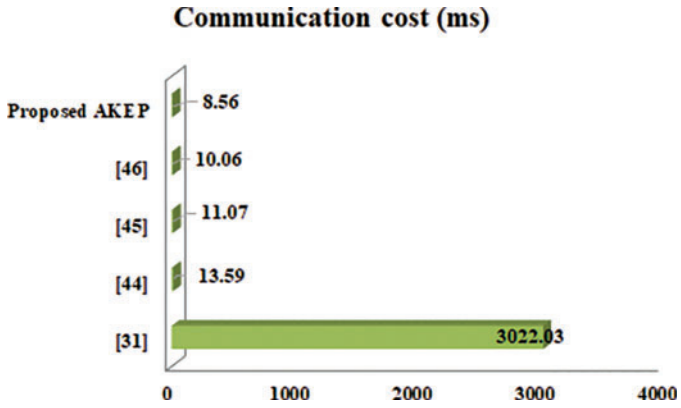



**Figure 3:** The total communication costs

## 6 Conclusion

In this paper, we propose a three-party password-based AKEP with client anonymity that uses Chebyshev chaotic maps and is more efficient and secure than previous procedures. Chebyshev chaotic maps are used to encrypt and decrypt data transmitted by either the client or the server to increase security and efficiency. Because it solely employs hash and XOR operations, we were able to show through security and performance studies that our AKEP is more effective and secure than others. To protect client identification when transferring data over an insecure public network, our AKEP may also offer client anonymity. In the future, we plan to provide efficient, lightweight, provably secure authentication protocols for the Internet of Things based on authenticated key exchange procedures using Chebyshev chaotic maps.

**Acknowledgement:** The authors would like to thank anonymous reviewers of Computers, Materials & Continua Journal for their careful and helpful comments. Prof. Monia Hadj Alouane-Turki extends her appreciation to the Deanship of Scientific Research at King Khalid University for funding this work through Small Groups. Project under grant number (RGP.1/257/43).

**Funding Statement:** The authors received no specific funding for this study.

**Conflicts of Interest:** The authors declare that they have no conflicts of interest to report regarding the present study.

## References

[1] L. Atzori, A. Iera and G. Morabito, "The internet of things: A survey," *Computer Networks*, vol. 54, no. 15, pp. 2787–2805, 2010.
[2] A. Valenzano, L. Durante and M. Cheminod, "Review of security issues in industrial networks," *IEEE Transactions on Industrial Informatics*, vol. 9, no. 1, pp. 277–293, 2013.
[3] V. C. Gungor and G. P. Hancke, "Industrial wireless sensor networks: Challenges, design principles and technical approaches," *IEEE Transactions on Industrial Electronics*, vol. 56, no. 10, pp. 4258–4265, 2009.
[4] D. Liu, M. C. Lee and D. Wu, "A node-to-node location verification method," *IEEE Transactions on Industrial Electronics*, vol. 57, no. 5, pp. 1526–1537, 2010.
[5] C. Chang and C. Lee, "A secure single sign-on mechanism for distributed computer networks," *IEEE Transactions on Industrial Electronics*, vol. 59, no. 1, pp. 629–637, 2012.
[6] G. Wang, J. Yu and Q. Xie, "Security analysis of a single sign-on mechanism for distributed computer networks," *IEEE Transactions on Industrial Informatics*, vol. 9, no. 1, pp. 294–302, 2013.
[7] L. Barolli and F. Xhafa, "JXTA-Overlay: A P2P platform for distributed, collaborative and ubiquitous computing," *IEEE Transactions on Industrial Electronics*, vol. 58, no. 6, pp. 2163–2172, 2010.
[8] Y. Huang, W. Lin and H. Li, "Efficient implementation of RFID mutual authentication protocol," *IEEE Transactions on Industrial Electronics*, vol. 59, no. 12, pp. 4784– 4791, 2012.
[9] B. Wang and M. Ma, "A server independent authentication scheme for RFID systems," *IEEE Transactions on Industrial Informatics*, vol. 8, no. 3, pp. 689–696, 2012.
[10] B. Fabian, T. Ermakova and C. Muller, "SHARDIS: A privacy enhanced discovery service for RFID-based product information," *IEEE Transactions on Industrial Informatics*, vol. 8, no. 3, pp. 707–718, 2012.
[11] T. Y. Chang, M. S. Hwang and W. P. Yang, "A communication-efficient three-party password authenticated key exchange protocol," *Information Sciences*, vol. 181, no. 1, pp. 217–226, 2011.
[12] T. Y. Chang, W. P. Yang and M. S. Hwang, "Simple authenticated key agreement and protected password change protocol," *Computers & Mathematics with Applications*, vol. 49, no. 5–6, pp. 703–714, 2005.
[13] T. F. Lee, T. Hwang and C. L. Lin, "Enhanced three-party encrypted key exchange without server public keys," *Computers & Security*, vol. 23, no. 7, pp. 571–577, 2004.

[14] S. W. Lee, H. S. Kim and K. Y. Yoo, "Efficient verifier-based key agreement protocol for three parties without server's public key," *Applied Mathematics and Computation*, vol. 167, no. 2, pp. 996–1003, 2005.
[15] D. He, Y. Chen and J. Chen, "Cryptanalysis and improvement of an extended chaotic maps-based key agreement protocol," *Nonlinear Dynamics*, vol. 69, no. 3, pp. 1149–1157, 2012.
[16] J. W. Lo, J. Z. Lee, M. S. Hwang and Y. P. Chu, "An advanced password authenticated key exchange protocol for imbalanced wireless networks," *Journal of Internet Technology*, vol. 11, no. 7, pp. 997–1004, 2010.
[17] J. W. Lo, S. C. Lin and M. S. Hwang, "A parallel password-authenticated key exchange protocol for wireless environments," *Information Technology and Control*, vol. 39, no. 2, pp. 146–151, 2010.
[18] S. M. Bellovin and M. Merritt, "Encrypted key exchange: Password-based protocols secure against dictionary attacks," in *Proc. of IEEE Computer Society Symp. on Security and Privacy*, Oakland, CA, USA, pp. 72–84, 1992.
[19] C. C. Lee, R. X. Chang and H. J. Ko, "Improving two novel three-party encrypted key exchange protocols with perfect forward secrecy," *International Journal of Foundations of Computer Science*, vol. 21, no. 6, pp. 979–991, 2010.
[20] C. C. Lee and Y. F. Chang, "On security of a practical three-party key exchange protocol with round efficiency," *Information Technology and Control*, vol. 37, no. 4, pp. 333–335, 2008.
[21] M. I. Hassan and A. Abdullah, "A new grid resource discovery framework," *The International Arab Journal of Information Technology*, vol. 8, no. 1, pp. 99–107, 2011.
[22] M. Abdalla and D. Pointcheval, "Interactive Diffie-Hellman assumptions with applications to password-based authentication," *Lecture Notes in Computer Science*, vol. 3570, pp. 341–356, 2005.
[23] R. Lu and Z. Cao, "Simple three-party key exchange protocol," *Computers & Security*, vol. 26, no. 1, pp. 94–97, 2007.
[24] B. H. T. Algubili, N. Kumar, H. Lu, A. A. Yassin, R. Boussada *et al.,* "EPSAPI: An efficient and provably secure authentication protocol for an IoT application environment," *Peer-to-Peer Networking and Applications*, vol. 15, pp. 2179–2198, 2022.
[25] M. Deng, J. Ma and F. Le, "Universally composable three party password-based key exchange protocol," *China Communications*, vol. 6, no. 3, pp. 150–154, 2009.
[26] W. Yuan, L. Hu, H. Li and J. Chu, "Offline dictionary attack on a universally composable three-party password-based key exchange protocol," *Procedia Engineering*, vol. 15, no. 11, pp. 1691–1694, 2011.
[27] H. F. Huang, "A simple three-party password-based key exchange protocol," *International Journal of Communication Systems*, vol. 22, no. 7, pp. 857–862, 2009.
[28] E. J. Yoon and K. Y. Yoo, "Cryptanalysis of a simple three-party password-based key exchange protocol," *International Journal of Communication Systems*, vol. 24, no. 4, pp. 532–542, 2011.
[29] E. J. Yoon and K. Y. Yoo, "Cryptanalysis of an efficient three-party password-based key exchange scheme," *Procedia Engineering*, vol. 29, no. 5, pp. 3972–3979, 2012.
[30] D. C. Lou and H. F. Huang, "Efficient three-party password-based key exchange scheme," *International Journal of Communication Systems*, vol. 24, no. 4, pp. 504–512, 2011.
[31] S. Wu, K. Chen and Y. Zhu, "Enhancements of a three-party password-based authenticated key exchange protocol," *International Arab Journal of Information Technology*, vol. 10, no. 3, pp. 215–221, 2013.
[32] Q. Zhang, P. Chaudhary, S. Kumari, Z. Kong and W. Liu, "Verifier-based anonymous password-authenticated key exchange protocol in the standard model," *Mathematical Biosciences and Engineering*, vol. 16, no. 5, pp. 3623–3640, 2019.
[33] Q. Shu, S. B. Wang, B. Hu and L. D. Han, "Verifier-based three-party password-authenticated key exchange protocol from ideal lattices," *Journal of Cryptologic Research*, vol. 8, no. 2, pp. 294–306, 2021.
[34] C. Peikert, "Lattice cryptography for the internet," in *Proc. of the Int. Workshop on Post-quantum Cryptography*, Waterloo, ON, Canada, pp. 197–219, 2014.
[35] R. M. Campello de Souza, H. M. A. de Oliveira Kauffman, A. N. Kauffman and A. J. A. Paschoal, "Trigonometry in finite fields and a new Hartley transform," in *Proc. of IEEE Int. Symp. Information Theory (ISIT'98)*, Cambridge, MA, USA, pp. 293, 1998.

[36] J. B. Lima, D. Panariob and R. M. Campello de Souza, "Public-key encryption based on Chebyshev polynomials over GF(q)," *Information Processing Letters*, vol. 111, pp. 51–56, 2010.
[37] J. B. Lima, R. M. Campello de Souza and D. Panariob, "Security of publickey cryptosystems based on Chebyshev polynomials over prime finite fields," in *Proc. IEEE Int. Symp. Information Theory (ISIT 2008)*, Toronto, ON, Canada, pp. 1843–1847, 2008.
[38] J. C. Mason and D. C. Handscomb, *Chebyshev polynomials*. Boca Raton, Florida: Chapman & Hall/CRC, 2003.
[39] C. Meshram, C. C. Lee, S. G. Meshram and A. Meshram, "OOS-SSS: An efficient online/offline subtree-based short signature scheme using chebyshev chaotic maps for wireless sensor network," *IEEE Access*, vol. 8, no. 1, pp. 80063–80073, 2020.
[40] C. Meshram, C. C. Lee, A. S. Ranadive, C. T. Li, S. G. Meshram *et al.,* "A subtree-based transformation model for cryptosystem using chaotic maps under cloud computing environment for fuzzy user data sharing," *International Journal of Communication Systems*, vol. 33, no. 7, pp. e4307, 2020.
[41] P. Bergamo, P. D'Arco, A. De Santis and L. Ko Carev, "Security of public-key cryptosystems based on Chebyshev polynomials," *IEEE Transactions on Circuits and Systems*, vol. 52, no. 7, pp. 1382–1393, 2005.
[42] J. Li, L. Wang, L. Wang, X. Wang, Z. Huang *et al.,* "Verifiable chebyshev maps-based chaotic encryption schemes with outsourcing computations in the cloud/fog scenarios," *Concurrency and Computation: Practice and Experience*, vol. 31, no. 5, pp. e4523, 2018.
[43] D. Kahrobaei, C. Koupparis and V. Shpilrain, "Public key exchange using matrices over group rings," *Groups, Complexity, Cryptology*, vol. 5, no. 1, pp. 11–97, 2013.
[44] L. Yanrong and Z. Dawei, "A chaotic-map-based password-authenticated key exchange protocol for telecare medicine information systems," *Security and Communication Networks*, vol. 2021, pp. 1–8, 2021. https://doi.org/10.1155/2021/7568538
[45] X. Guo and J. Zhang, "Secure group key agreement protocol based on chaotic hash," *Information Sciences*, vol. 180, no. 20, pp. 4069–4074, 2010.
[46] T. -F. Lee, "Enhancing the security of password authenticated key agreement protocols based on chaotic maps," *Information Sciences*, vol. 290, pp. 63–71, 2015.
[47] C. Meshram, S. G. Meshram, R. W. Ibrahim, H. A. Jalab, S. S. Jamal *et al.,* "Conformal chebyshev chaotic maps based remote user password authentication protocol using smart card," *Complex & Intelligent Systems*, vol. 8, pp. 973–987, 2022.
[48] M. B. Algehawi and A. Samsudin, "A new identity-based encryption (IBE) scheme using extended Chebyshev polynomial over finite fields Zp," *Physics Letters A*, vol. 374, pp. 4670–4674, 2010.
[49] M. H. Ibrahim, S. Kumari, A. K. Das, M. Wazid and V. Odelu, "Secure anonymous mutual authentication for star two-tier wireless body area networks," *Computer Methods and Programs in Biomedicine*, vol. 135, no. 1, pp. 37–50, 2016.
[50] C. Meshram, A. L. Imoize, S. S. Jamal, P. Tambare, A. R. Alharbi *et al.,* "An efficient three-factor authenticated key agreement technique using FCM under HC-IoT architectures," *Computers, Materials & Continua*, vol. 72, no. 1, pp. 1373–1389, 2022.
[51] C. Meshram, A. L. Imoize, S. S. Jamal, A. Aljaedi and A. R. Alharbi, "SBOOSP for massive devices in 5G WSNs using conformable chaotic maps," *Computers, Materials & Continua*, vol. 71, no. 3, pp. 4591–4608, 2022.