An Energy-Efficient Authentication Scheme Based on Chebyshev Chaotic Map for Smart Grid Environments

Liping Zhang, Yue Zhu, Wei Ren, *Member, IEEE*, Yinghan Wang, Kim-Kwang Raymond Choo, *Senior Member, IEEE*, and Neal N. Xiong, *Senior Member, IEEE*

***Abstract*—Electric vehicle charging is becoming more commonplace, but a number of challenges remain. For example, the wireless communications between vehicle users and aggregators can be subject to exploitation and hence, several authentication schemes have been designed to support varying levels of privacy protection. However, there are a number of limitations observed in existing authentication schemes, and examples include lack of anonymity and not considering charging peak in their design (and consequently, not meeting low energy consumption requirement in smart grid environments). More recently, there have been attempts to utilize Chevyshev chaotic map in the design of authentication mechanism, with the aims of reducing computational costs yet achieving high security. However, the security requirements of Chebyshev polynomials pose new challenges to the construction of Chebyshev chaotic maps-based authentication schemes. To solve these limitations, we propose an efficient Chebyshev polynomials algorithm by adopting a square matrix-based binary exponentiation algorithm to provide secure and efficient Chebyshev polynomial computation. We further construct an energy-efficient authentication and key negotiation scheme for the smart grid environments based on the proposed algorithm. Compared with five other competing schemes, our proposed authentication scheme achieves reduced computational and communication costs. In addition, the ProVerif tool is used to analyze the security of our proposed authentication scheme. The results show that the proposed scheme outperforms these five other schemes in terms of computation and communication overheads while achieving privacy preserving.**

***Index Terms*—Authentication, Chebyshev chaotic maps, key agreement, privacy protection, smart grid environment.**

Manuscript received March 18, 2021; revised April 19, 2021; accepted May 3, 2021. Date of publication May 7, 2021; date of current version November 19, 2021. This work was supported in part by the National Natural Science Foundation of China under Grant 61303237 and Grant 61972366; in part by the Open Research Project of the Hubei Key Laboratory of Intelligent Geo-Information Processing under Grant KLIGIP-2019B09; in part by the Foundation of Key Laboratory of Network Assessment Technology, Chinese Academy of Sciences under Grant KFKT2019-003; in part by the Foundation of Henan Key Laboratory of Network Cryptography Technology under Grant LNCT2020-A01; and in part by the Foundation of Guizhou Provincial Key Laboratory of Public Big Data under Grant 2018BDKFJJ009, Grant 2019BDKFJJ003, and Grant 2019BDKFJJ011. Kim-Kwang Raymond Choo was supported only by the Cloud Technology Endowed Professorship. *(Corresponding author: Wei Ren.)*

Liping Zhang, Yue Zhu, and Yinghan Wang are with the School of Computer Science, China University of Geosciences, Wuhan 430074, China.

Wei Ren is with the School of Computer Science, China University of Geosciences, Wuhan 430074, China, also with the Henan Key Laboratory of Network Cryptography Technology, Zhengzhou, China, and also with the Key Laboratory of Network Assessment Technology, Institute of Information Engineering, Chinese Academy of Sciences, Beijing 100093, China (e-mail: weirencs@cug.edu.cn).

Kim-Kwang Raymond Choo is with the Department of Information Systems and Cyber Security, Department of Electrical and Computer Engineering, and Department of Computer Science, The University of Texas at San Antonio (UTSA), San Antonio, TX 78249 USA, and also with the UniSA STEM, University of South Australia, Adelaide, SA 5095, Australia.

Neal N. Xiong is with Department of Mathematics and Computer Science, Northeastern State University, Tahlequah, OK 74464 USA.

Digital Object Identifier 10.1109/JIOT.2021.3078175

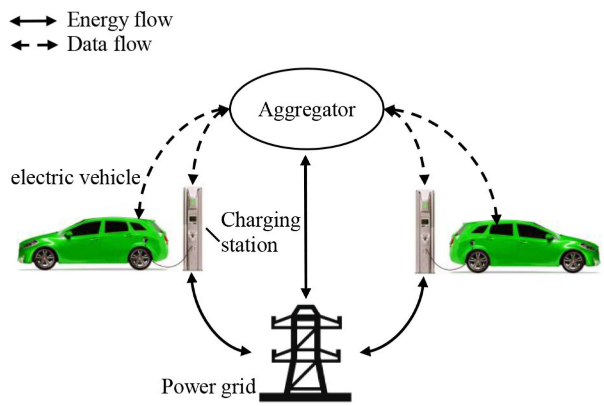


Fig. 1. -1System model of a basic V2G network.

I. Introduction

SMART grids (SGs) are generally more efficient, secure, and reliable in comparison to traditional grid infrastructures. SGs underpin many other applications, such as vehicle-to-grid (V2G) networks. The latter is an emerging application, particularly in a smart city/nation context, due to the potential for electric vehicles (EVs) to (significantly) reduce pollution by using renewable resources. As shown in Fig. 1, a basic V2G network comprises three entities, namely power grid (PG), aggregator (AGT), and EVs. The PG generates electricity from new renewable sources, such as solar and wind, and then sends the resulting electricity to the charging stations.

As an intermediary between EVs and the PG, the AGT monitors and collects the EVs' current state, optimizes and adjusts EVs' charging plan, and minimizes the charging energy cost of EVs [1]. When an EV owner (EVO) intends to recharge the vehicle, he/she first logs in to the system

2327-4662 © 2021 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission.
See https://www.ieee.org/publications/rights/index.html for more information.

Authorized licensed use limited to: China University of Geosciences Wuhan Campus. Downloaded on September 06,2023 at 05:52:07 UTC from IEEE Xplore. Restrictions apply.

using a smartcard and credentials through the EV's onboard unit [2]. Then, the EV will send a request to AGT to establish communications. Subsequently, the EVO can complete the payment by interacting with the charging station. However, the wireless communication in V2G networks can be subject to exploitation [3], [4], such as message modification attacks and man-in-the-middle attacks. In other words, sensitive information, such as EVO's identity, EV's location, and EV's route, may be leaked, and such information can be used to facilitate other nefarious activities (e.g., identity theft, financial fraud, and covert surveillance). This reinforces the importance of a secure and efficient authentication solution.

Several authentication schemes have been proposed to achieve secure communication in V2G and SG environments [1], [5]–[11], such as those based on public-key cryptography. However, authentication schemes based on public-key cryptography tend not to be energy friendly. In practice, we also have to consider peak and off-peak charging periods, which may cause a spike in demand and shortage of charging stations [12], [13]. Such a feature is not typically considered in existing authentication schemes. Therefore, how to design a secure and efficient authentication and key negotiation scheme for SGs, and in particular V2G networks, remains a challenging task.

We posit the potential of using Chebyshev polynomials in designing authentication and key negotiation schemes, with minimal computational costs and high security. However, most existing public-key algorithms based on Chebyshev polynomials to deal with real numbers is not secure [14]. To make the public key algorithm more secure and practical, Kocarev *et al.* [15] extended Chebyshev polynomials from real fields to finite fields and finite rings. Chen *et al.* [16] subsequently proved that the Chebyshev polynomials $T_n(x)$ mod N is safe when modulus N is a strong prime number satisfying $N - 1 = 2p_1$ and $N + 1 = 2p_2$, where p_1 and p_2 are also prime numbers. Therefore, modulus N should be carefully selected to ensure that Chebyshev polynomials can produce sequences of sufficient period to resist violent attacks. However, in existing Chebyshev polynomials algorithms, such as the algorithms adopted in [17] and [18], the parameter n of Chebyshev polynomials $T_n(x)$ mod N is constructed with small primes, which compound the challenge of designing secure Chebyshev chaotic map-based authentication schemes.

These challenges motivate us to design a practical Chebyshev polynomial algorithm and use it as a building block in the design of a secure and efficient authentication scheme for SG environments.

1) Specifically, we propose an efferent Chebyshev polynomial algorithm that adopts a square matrix-based binary exponentiation algorithm to realize secure and practical Chebyshev polynomial computation. Our proposed algorithm guarantees the security requirements that are proven by Chen *et al.* [16]. Also, our experimental results demonstrate that the proposed algorithm is an efficient Chebyshev polynomial algorithm.
2) Using the proposed Chebyshev polynomials algorithm as a building block, we further construct an energy-efficient authentication scheme for SG environments. Our scheme realizes fast authentication and key negotiation with anonymity.

In the next section, we will briefly review the related literature. The mathematical background of Chebyshev polynomial is described in Section III. In Sections IV and V, we will present our proposed Chebyshev polynomial algorithm and authentication scheme. In Section VI, using ProVerif (an automatic verifier) we demonstrate that our authentication scheme can resist known attacks. We also remark that our proposed authentication scheme is lightweight since only efficient Chebyshev polynomials and hash functions are adopted during the authentication and key negotiations process. A comparative summary in Section VII also shows that the proposed authentication scheme is more efficient than five other competing schemes [1], [19]–[22]. Finally, we conclude this article in the last section.

II. Related Work

Over the years, a large number of authentication schemes based on public-key cryptography for SGs have been proposed. Examples include those of Wu and Zhou [23], Xia and Wang [24], Tsai and Lo [25], etc. To further enhance security while reducing computational costs, there have been attempts to utilize elliptic curve cryptography (ECC) in the design of authentication schemes, such as in the approaches of He *et al.* [26], Odelu *et al.* [20], Abbasinezhad-Mood and Nikooghadam [27], Mahmood *et al.* [21], and Kumar *et al.* [22]. However, the elliptic curve point multiplication operations in these schemes are time-consuming operations, particularly in SG environments.

As we discussed in the preceding section, there have also been attempts to use Chebyshev polynomials (a lightweight operation) in the design of authentication schemes. Chaotic maps based authentication schemes have been designed for various environments, such as multiserver [28], [29], isolated smart meters [30], and point-of-care systems [31]. Recently, Abbasinezhad-Mood *et al.* [1] adopted Chebyshev chaotic maps to design an authentication scheme for the V2G communications. While their scheme reduces computational costs in theory, the scheme only executes each cryptography operation independently on the device and then calculate a theoretical time as the execution time of their scheme, without considering other operational factors in a real-world deployment.

III. Preliminaries

In this section, we review the basic concepts of Chebyshev chaotic and the corresponding difficult problems associated with it.

Definition 1 (Chebyshev Chaotic Map): Let n be an integer and $x \in [-1, 1]$, the Chebyshev polynomial is defined as (1) or (2) [32]–[35]

$$T_n(x) = \cos(n \cos^{-1}(x)) \tag{1}$$

$$T_n(x) = 2xT_{n-1}(x) - T_{n-2}(x); n \geq 2,\ T_0(x) = 1,\ T_1(x) = x. \tag{2}$$

Authorized licensed use limited to: China University of Geosciences Wuhan Campus. Downloaded on September 06,2023 at 05:52:07 UTC from IEEE Xplore. Restrictions apply.

Definition 2 (Semigroup Property): One of the most important property of Chebyshev polynomial is the semigroup property, which is shown as

$$T_u(T_v(x)) = T_{uv}(x) = T_v(T_u(x)). \quad (3)$$

Zhang [36] demonstrated that the semigroup property of Chebyshev polynomials also holds, when Chebyshev polynomial domain is defined on intervals $(-\infty, +\infty)$. The enhanced Chebyshev polynomial is defined as (4), where p is a large prime number

$$T_n(x) = (2xT_{n-1}(x) - T_{n-2}(x)) \bmod p, \ n \geq 2, \ x \in (-\infty, +\infty). \quad (4)$$

Definition 3 (Chaotic Map-Based Discrete Logarithm Problem (CMBDLP) [35], [37], [38]): Given x and y, it is almost impossible to find the integer v, such that $T_v(x) = y$. The probability that an adversary A can solve the CMBDLP is defined as $\mathrm{Adv}_A^{\mathrm{CMBDLP}}(p) = \Pr[A(x, y) = v : v \in Z_p^*, y = T_v(x) \bmod p]$.

Definition 4 (CMBDLP Assumption [35], [37], [38]): For any probabilistic polynomial time-bounded adversary A, $\mathrm{Adv}_A^{\mathrm{CMBDLP}}(p)$ is negligible, that is, $\mathrm{Adv}_A^{\mathrm{CMBDLP}}(p) < \varepsilon$.

Definition 5 (Chaotic Map-Based Diffie–Hellman Problem (CMBDHP) [35], [37], [38]): Given x, $T_u(x)$ and $T_v(x)$, it is almost impossible to find $T_{uv}(x)$. The probability that a polynomial time-bounded adversary A can solve the CMBDHP is defined as $\mathrm{Adv}_A^{\mathrm{CMBDHP}}(p) = \Pr[A(x, T_u(x) \bmod p, T_v(x) \bmod p = T_{uv}(x) \bmod p : u, v \in Z_p^*]$.

Definition 6 (CMBDHP Assumption [35], [37], [38]): For any probabilistic polynomial time-bounded adversary A, $\mathrm{Adv}_A^{\mathrm{CMBDHP}}(p)$ is negligible, that is, $\mathrm{Adv}_A^{\mathrm{CMBDHP}}(p) < \varepsilon$.

IV. Our Proposed Chebyshev Polynomial Algorithm

In this section, we describe our proposed Chebyshev polynomial algorithm. To satisfy the security requirements and reduce time complexity, we adopt a binary exponentiation algorithm based on a square matrix to compute Chebyshev polynomials. Furthermore, we employ the following matrices instead of recursive relationships to define Chebyshev polynomials:

$$\begin{bmatrix} T_{n+1}(x) \\ T_n(x) \end{bmatrix} = \begin{bmatrix} 2x & -1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} T_n(x) \\ T_{n-1}(x) \end{bmatrix} \bmod q$$

$$\begin{bmatrix} T_{n+1}(x) \\ T_n(x) \end{bmatrix} = \begin{bmatrix} 2x & -1 \\ 1 & 0 \end{bmatrix}^2 \begin{bmatrix} T_{n-1}(x) \\ T_{n-2}(x) \end{bmatrix}$$

$$\Rightarrow \begin{bmatrix} 2x & -1 \\ 1 & 0 \end{bmatrix}^n \begin{bmatrix} T_1(x) \\ T_0(x) \end{bmatrix} \bmod q.$$

The nth power modulus q of the matrix $\begin{bmatrix} 2x & -1 \\ 1 & 0 \end{bmatrix}$ can be solved by the binary power algorithm in polynomial time. The detailed steps of the proposed algorithm to compute $T_n(x)$ mod N are shown in Fig. 2, where $[n]$ denotes the integer part of n.

To meet the security requirements, in our proposed Chebyshev polynomial algorithm, n is a large prime number, and modulus N is a strong prime number satisfying $N - 1 = 2p_1$ and $N + 1 = 2p_2$. Therefore, our proposed algorithm is safe according to [16]. Furthermore, we have implemented the Chebyshev polynomial operations on Intel Pentium CPU G850 to investigate the practicability of our proposed algorithm. Table I illustrates that the proposed algorithm is an efficient Chebyshev polynomial algorithm. Furthermore, we can conclude from Table I that the execution time of Chebyshev polynomial increases with the parameter n increases. The above conclusion also shows that the calculation method of execution time adopted in [1] is inappropriate.

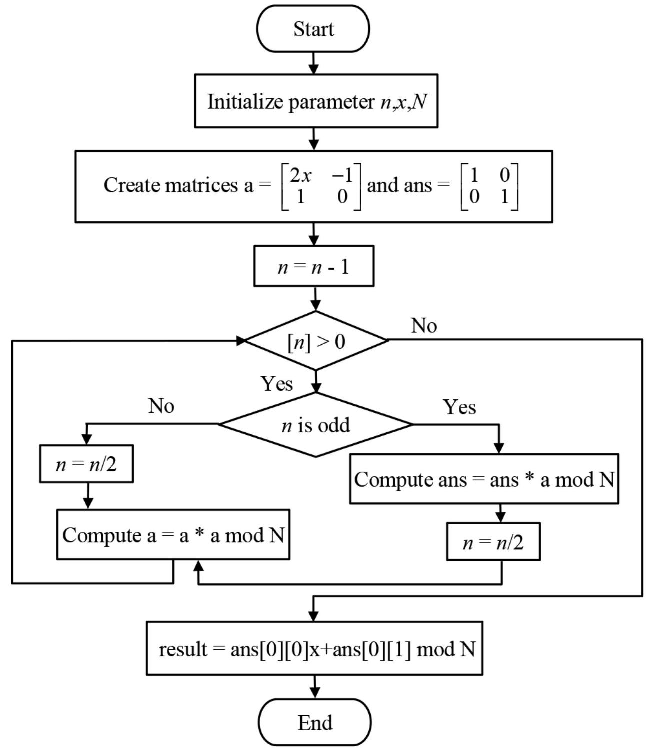


Fig. 2. Flowchart of Chebyshev polynomial algorithm.

TABLE I
EXECUTION TIMES OF CHEBYSHEV POLYNOMIAL

The bits of parameter n	Execution time
128bits	0.184628ms
160bits	0.237291ms
256bits	0.379685ms
512bits	0.752449ms

V. Our Proposed Authentication Scheme

Using the proposed Chebyshev polynomial algorithm described in Section IV as a building block, we will now present our lightweight authentication scheme. The latter consists of four phases, namely: system setup phase, registration phase, login phase, and authentication phase. First, the trusted authority (TA) generates some system parameters in the system setup phase. Then, in the registration phase, TA completes registration of the electric vehicle (EV_i) with a smartcard and the aggregator (AGT_j) using a secure channel. Finally, in the authentication phase, EV and AGT authenticate each other

Authorized licensed use limited to: China University of Geosciences Wuhan Campus. Downloaded on September 06,2023 at 05:52:07 UTC from IEEE Xplore. Restrictions apply.

TABLE II
NOTATIONS AND DEFINITIONS

Notation	Definition
ID_i	Identity of the i^{th} electric vehicle
ID_j	Identity of the j^{th} aggregator
x	The seed of Chebyshev polynomial
p	Large prime number
$h(\cdot)$	Secure hash function
$r_i, r_j, r_u, r_s, r_1, r_2$	high-entropy random numbers
k, pub_{TA}	Private/public key of *TA*
k_i, k_j	Private key of EV_i and AGT_j
SK_{ij}, SK_{ji}	Session key of EV_i and AGT_j
$\|\|$	Concatenation operation
$\oplus$	Exclusive-or operation

and establish session keys for future communications. Some notations employed in this article are described in Table II.

A. System Setup Phase

In this phase, trusted authority (TA) selects system parameters and generates its key pairs. Meanwhile, TA completes the registration of each aggregator AGT_j before the deployment. The detailed steps are given as follows.

Step 1: The trusted authority first chooses a large prime integer p, and then selects a high entropy random integer $x \in Z_n^*$ as the seed of Chebyshev polynomial.

Step 2: TA chooses a high entropy random integer $k \in Z_n^*$ as its private key, and then calculatesc its corresponding public key $\mathrm{pub}_{\mathrm{TA}}$ as in (5). Next, the TA generates its identity $\mathrm{ID}_{\mathrm{TA}}$ and calculates a pseudoidentity for itself using the value $\mathrm{ID}_{\mathrm{TA}}$ and its private key k as (6)

$$\mathrm{pub}_{\mathrm{TA}} = T_k(x) \bmod p \tag{5}$$

$$\mathrm{RID}_{\mathrm{TA}} = h(\mathrm{ID}_{\mathrm{TA}}||k). \tag{6}$$

Step 3: The trusted authority TA chooses a collision-resistant hash functions $h() : \{0,1\}^* \longrightarrow \{0,1\}^l$. Next, it publishes system parameters $\{p, x, \mathrm{pub}_{\mathrm{TA}}, h()\}$ and keeps its privacy key k secretly.

B. Registration Phase

When an electric vehicle EV_i wants to access the aggregator AGT_j, it needs to perform the following registration process. In this phase, a smartcard is issued for each electric vehicle owner and the communication channels are supposed to be secure.

Step 1: First, the aggregator AGT_j generates its identity ID_j, and selects a random integer $r_j \in Z_n^*$. Then it calculates its pseudoidentity RID_j as in (7) and sends it to the trusted authority TA in a secure channel. Next, the TA selects a high entropy random integer $r_s \in Z_n^*$ and computes Q_j as in (8) using this integer, the received value RID_j, its public key $\mathrm{pub}_{\mathrm{TA}}$ and private key k. Then TA further adopts the computed value Q_j, random integer r_s and its private key k to calculate the signature s_j as (9). After that, it sends the message $\{\mathrm{RID}_{\mathrm{TA}}, Q_j, s_j\}$ to aggregator AGT_j in a secure channel. Subsequently, aggregator AGT_j computes its private key k_j as (10) using the signature s_j received from the TA. Finally, the aggregator AGT_j stores the information $\{\mathrm{RID}_{\mathrm{TA}}, \mathrm{RID}_j, Q_j, k_j\}$ in its memory secretly. When this step is finished, the registration process of the aggregator AGT_j on the TA is completed

$$\mathrm{RID}_j = h\big(\mathrm{ID}_j \oplus r_j\big) \tag{7}$$

$$Q_j = T_{\mathrm{RID}_j} T_{h(r_s \oplus k)}\big(\mathrm{pub}_{\mathrm{TA}}\big) \tag{8}$$

$$s_j = h\big(\mathrm{RID}_j||Q_j\big)h(r_s \oplus k)k \tag{9}$$

$$k_j = h\big(\mathrm{ID}_j \oplus r_j\big)s_j. \tag{10}$$

Step 2: The electric vehicle owner (EVO) freely chooses his/her identity ID_i and password PW_i. Then, it selects a high entropy random integer $r_i \in Z_n^*$ and calculates its pseudoidentity RID_i as in (11). It also adopts its identity ID_i and password PW_i to compute RPW_i as in (12). After that, EVO sends message $\{\mathrm{ID}_i, \mathrm{RID}_i, \mathrm{RPW}_i\}$ to the trusted authority TA in a secure channel. Next, the trusted authority TA selects a high entropy random integer $r_u \in Z_n^*$ and then uses this integer, EVO's pseudoidentity RID_i, public key $\mathrm{pub}_{\mathrm{TA}}$ and private key k to obtain Q_i as in (13). Then, the TA further calculates the signature s_i as (14) via the computed value Q_i, pseudoidentity RID_i, random integer r_u and private key k. Next, the TA adopts its private key k and the computed value RPW_i to generate Y as in (15). Then TA sends $\{\mathrm{ID}_i, \mathrm{RPW}_i\}$ to the relevant AGT_j. After receiving the message, the AGT_j computes A_i as (16), Z as (17) and M_i as (18). Afterward, AGT_j sends $\{Z, M_i, \mathrm{RID}_j, Q_j\}$ to the TA. After that, TA writes $\{Z, M_i, \mathrm{RID}_j, Q_j, \mathrm{RID}_{\mathrm{TA}}, Y, s_i\}$ into the smartcard and sends the smartcard to EVO using a secure way. When EVO receives the smartcard, it adopts Y stored in the smartcard and its identity ID_i to compute I as (19). And then it computes the private key k_i as (20) using its privacy information $\{\mathrm{ID}_i, \mathrm{PW}_i, r_i\}$ and the signature s_i of the TA. Then, the EVO stores the information $\{k_i\}$ and replaces the Y with I in the memory of his/her smartcard. Finally, the memory of the smartcard contains $\{\mathrm{RID}_{\mathrm{TA}}, I, k_i, Z, M_i, \mathrm{RID}_j, Q_j\}$. After this step, the EV/EVO finishes the registration at the TA

$$\mathrm{RID}_i = h(r_i \oplus \mathrm{ID}_i) \tag{11}$$

$$\mathrm{RPW}_i = h(\mathrm{ID}_i \oplus \mathrm{PW}_i) \tag{12}$$

$$Q_i = T_{\mathrm{RID}_i} T_{h(r_u \oplus k)}\big(\mathrm{pub}_{\mathrm{TA}}\big) \tag{13}$$

$$s_i = h(\mathrm{RID}_i||Q_i)h(r_u \oplus k)k \tag{14}$$

$$Y = T_{\mathrm{RPW}_i} T_k(x) \tag{15}$$

$$A_i = h\big(\mathrm{ID}_i||k_j\big) \tag{16}$$

$$Z = \mathrm{RPW}_i \oplus A_i \tag{17}$$

$$M_i = \mathrm{ID}_i \oplus h\big(k_j\big) \tag{18}$$

$$I = T_{\mathrm{ID}_i}(Y) \tag{19}$$

$$k_i = h(r_i \oplus \mathrm{ID}_i||\mathrm{PW}_i)s_i. \tag{20}$$

C. Login Phase

In this phase, the registered EVO makes a login request to the aggregator AGT_j.

Authorized licensed use limited to: China University of Geosciences Wuhan Campus. Downloaded on September 06,2023 at 05:52:07 UTC from IEEE Xplore. Restrictions apply.

Step 1: The EVO inserts the smartcard into EV_i and then inputs the identity ID_i and password PW_i.

Step 2: Then the electric vehicle EV_i computes RPW_i as (21) using the inputted identity ID_i and PW_i and computes I_0 as (22) to check whether the value of I_0 and I are equal. If true, electric vehicle EV_i selects a high entropy random integer $r_1 \in Z_n^*$ and calculates C_1 as (23). Then EV_i computes A_i as (24) and C_2 as (25). Finally, the electric vehicle EV_i sends the login request $\{C_1, C_2, M_i\}$ to the corresponding aggregator AGT_j via a public channel

$$\mathrm{RPW}_i = h(\mathrm{ID}_i \oplus \mathrm{PW}_i) \tag{21}$$
$$I_0 = T_{\mathrm{ID}_i}\big(T_{\mathrm{RPW}_i}(\mathrm{pub}_{\mathrm{TA}})\big) \tag{22}$$
$$C_1 = T_{r_1}(x) \tag{23}$$
$$A_i = Z \oplus \mathrm{RPW}_i \tag{24}$$
$$C_2 = h\big(\mathrm{ID}_i||\mathrm{RID}_j||A_i||C_1\big). \tag{25}$$

D. Authentication Phase

To achieve authentication and key negotiation, some messages are required to transmit between electric vehicle EV_i and accessed aggregator AGT_j during the authentication phase. The detailed steps are shown as follows.

Step 1: When AGT_j receives the login request $\{C_1, C_2, M_i\}$, it first gets the identity of EV_i as (26) using its secret key k_j and the received M_i. Next, it further computes A_i' as (27), C_2' as (28) and checks whether the equation $C_2' = C_2$ holds. If true, it selects a random integer $r_2 \in Z_n^*$ and computes C_3 as (29). And the aggregator AGT_j further computes X as (30). Finally, it generates an authentication message Auth_s as (31) and sends message $\{C_3, \mathrm{Auth}_s\}$ to the EV_i via a public channel

$$\mathrm{ID}_i = M_i \oplus h\big(k_j\big) \tag{26}$$
$$A_i' = h\big(\mathrm{ID}_i||k_j\big) \tag{27}$$
$$C_2' = h\big(\mathrm{ID}_i||\mathrm{RID}_j||A_i'||C_1\big) \tag{28}$$
$$C_3 = T_{r_2}\big(Q_j\big) \tag{29}$$
$$X = T_{r_2 k_j}(C_1) = T_{r_1 r_2 k_j}(x) \tag{30}$$
$$\mathrm{Auth}_s = h\big(X||A_i'||\mathrm{ID}_i||\mathrm{RID}_j||C_3\big). \tag{31}$$

Step 2: When the electric vehicle EV_i receives responding message $\{C_3, \mathrm{Auth}_s\}$ from aggregator AGT_j, it uses the $\{\mathrm{RID}_j, Q_j\}$ and C_3 to obtain X' as (32). Next, the electric vehicle EV_i computes $\mathrm{Auth}_s{}'$ as (33) and compares it with Auth_s. If they are equivalent, it computes C_4 as (34). After that, the electric vehicle EV_i uses the computed value X' and C_4 to generate its authentication message Auth_u as (35). Finally, the electric vehicle EV_i computes the session key SK_{ij} as (36) and sends the message $\{C_4, \mathrm{Auth}_u\}$ to the aggregator AGT_j

$$X' = T_{r_1}\Big(T_{h(\mathrm{RID}_j||Q_j)}(C_3)\Big) = T_{r_1 h(\mathrm{RID}_j||Q_j) r_2}\big(Q_j\big) \tag{32}$$
$$\mathrm{Auth}_s{}' = h\big(X'||A_i||\mathrm{ID}_i||\mathrm{RID}_j||C_3\big) \tag{33}$$
$$C_4 = h\big(A_i||X'\big) \tag{34}$$
$$\mathrm{Auth}_u = h\big(X'||C_4\big) \tag{35}$$
$$\mathrm{SK}_{ij} = h\big(\mathrm{RID}_{\mathrm{TA}}||X'||A_i\big). \tag{36}$$

Step 3: When aggregator AGT_j obtains the message from the EV_i, it first computes $\mathrm{Auth}_u{}'$ as (37). And then the AGT_j checks whether the values of $\mathrm{Auth}_u{}'$ and Auth_u are equal. If equal, The aggregator AGT_j obtains the shared session key $\mathrm{SK}_{ji}(= \mathrm{SK}_{ij})$ as (38)

$$\mathrm{Auth}_u' = h(X||C_4) = h\Big(T_{r_2 k_j}(C_1)||C_4\Big) \tag{37}$$
$$\mathrm{SK}_{ji} = h\big(\mathrm{RID}_{\mathrm{TA}}||X||A_i'\big). \tag{38}$$

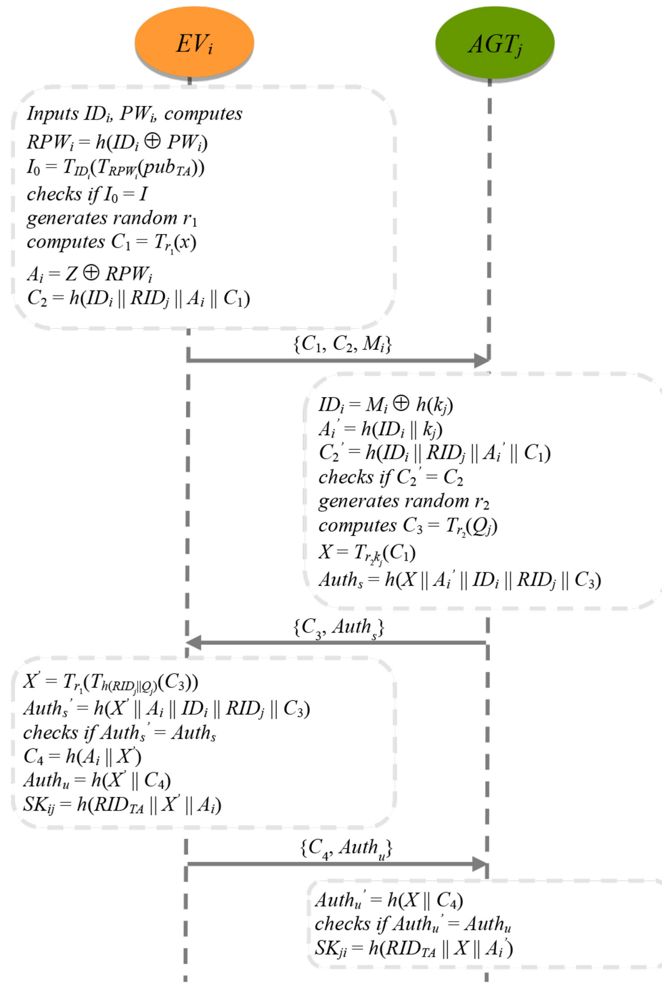


Fig. 3. Illustration of login and authentication phases.

Finally, the electric vehicle EV_i and the aggregator AGT_j achieve mutual authentication and key negotiation. The login and authentication processes also are shown in Fig. 3.

Suppose that the EV_i and the AGT_j are legal. The SK_{ij} is the session key generated by the EV_i and the SK_{ji} is the session key computed by the AGT_j. Now we prove that the equation $\mathrm{SK}_{ij} = \mathrm{SK}_{ij}$ is held in our proposed scheme.

Proof:

$$\begin{aligned}
\mathrm{SK}_{ij} &= h\big(\mathrm{RID}_{\mathrm{TA}}||X'||A_i\big)\\
&= h\Big(\mathrm{RID}_{\mathrm{TA}}||T_{r_1}\Big(T_{h(\mathrm{RID}_j||Q_j)}(C_3)\Big)||h\big(\mathrm{ID}_i||k_j\big)\\
&= h\Big(\mathrm{RID}_{\mathrm{TA}}||T_{r_1}\Big(T_{h(\mathrm{RID}_j||Q_j)}\big(T_{r_2}\big(Q_j\big)\big)\Big)||h\big(\mathrm{ID}_i||k_j\big)\\
&= h\Big(\mathrm{RID}_{\mathrm{TA}}||T_{r_1 r_2 h(\mathrm{RID}_j||Q_j)} T_{\mathrm{RID}_j} T_{(r_s \oplus k)}\big(\mathrm{pub}_{\mathrm{TA}}\big)||A_i'\Big)\\
&= h\Big(\mathrm{RID}_{\mathrm{TA}}||T_{r_1 r_2 h(\mathrm{RID}_j||Q_j) h(\mathrm{ID}_j \oplus r_j) h(r_s \oplus k) k}(x)||A_i'\Big)
\end{aligned}$$

Authorized licensed use limited to: China University of Geosciences Wuhan Campus. Downloaded on September 06,2023 at 05:52:07 UTC from IEEE Xplore. Restrictions apply.

```
(************************* EVi *************************)
let pEVi =
in (c, (C3 : bitstring, Auths : bitstring));
let xX = Cheb(r1,Cheb(HashFunTwo(RIDj,Qj),C3)) in
let xAuths = HashFunFive(xX,Ai,IDi,RIDj,C3) in
if (xAuths = Auths) then
let C4 = HashFunTwo(Ai,xX) in
let Authu = HashFunTwo(xX,C4) in
let SKij = HashFunThree(RIDTA,xX,Ai) in
event startEVi;
out (c,(C4,Authu));
event endEVi;
0.

(************************* AGTj ************************)
let pAGTj =
new r2: bitstring;
in (c, (C1 : bitstring, C2 : bitstring, Mi : bitstring));
let IDi = DXORFun(Mi,HashFunOne(kj)) in
let xAi = HashFunTwo(IDi,kj) in
let xC2 = HashFunFour(IDi,RIDj,xAi,C1) in
if (xC2 = C2) then
let C3 = Cheb(r2,Qj) in
let X = Cheb(MulFun(r2,kj),C1) in
let Auths = HashFunFive(X,xAi,IDi,RIDj,C3) in
event startAGTj;
out (c,(C3,Auths));
in (c, (C4 : bitstring, Authu : bitstring));
let xAuthu = HashFunTwo(X,C4) in
if (xAuthu = Authu) then
let SKji = HashFunThree(RIDTA,X,xAi) in
event endAGTj;
0.

process
( (!pEVi) | (!pAGTj) )
```

Fig. 4. Authentication phase of the EV_i and the AGT_j.

$$
\begin{aligned}
&= h\left(\mathrm{RID}_{\mathrm{TA}}||T_{r_1 r_2 h(\mathrm{ID}_j \oplus r_j) s_j}(x)||A_i'\right)\\
&= h\left(\mathrm{RID}_{\mathrm{TA}}||T_{r_1 r_2 k_j}(x)||A_i'\right)\\
&= h\left(\mathrm{RID}_{\mathrm{TA}}||T_{r_2 k_j}(C_1)||A_i'\right)\\
&= h\left(\mathrm{RID}_{\mathrm{TA}}||X||A_i'\right)\\
&= \mathrm{SK}_{ji}.
\end{aligned}
$$

■

VI. Security Analysis

In this section, we adopt an automatic verifier named ProVerif to analyze the security of our proposed scheme. Moreover, several possible attacks are discussed in Section VI-B.

A. Automatic Formal Verification of Security Using ProVerif

In this section, we demonstrate the security of the proposed scheme using a widely accepted automatic protocol verifier named ProVerif [39]. ProVerif can be utilized to verify observational equivalences, correspondence assertions, and reachability properties. Specifically, we can validate the resistance of cryptographic protocols against impersonation attacks, modification attacks, and replay attacks by launching injective correspondence assertion queries. Moreover, by using observational equivalence queries, some security properties, such as identity guessing attacks can be verified via ProVerif. Furthermore, by making reachability queries, both the anonymity feature and the secrecy of the session key can be checked. Significantly,

```
RESULT not attacker(ki[]) is true.                                  (1)
RESULT not attacker(kj[]) is true.                                  (2)
RESULT not attacker(SKij[]) is true.                                (3)
RESULT not attacker(SKji[]) is true.                                (4)
RESULT not attacker(IDi[]) is true.                                 (5)
RESULT not attacker(IDj[]) is true.                                 (6)
RESULT inj-event(endAGTj) ==> inj-event(startAGTj) is true.         (7)
RESULT inj-event(endEVi) ==> inj-event(startEVi) is true.           (8)
```

Fig. 5. Results from the ProVerif(1).

```
RESULT not attacker(ki[]) is false.                                 (1)
RESULT not attacker(kj[]) is false.                                 (2)
RESULT not attacker(SKij[]) is true.                                (3)
RESULT not attacker(SKji[]) is true.                                (4)
RESULT not attacker(IDi[]) is true.                                 (5)
RESULT not attacker(IDj[]) is true.                                 (6)
RESULT inj-event(endAGTj) ==> inj-event(startAGTj) is true.         (7)
RESULT inj-event(endEVi) ==> inj-event(startEVi) is true.           (8)
```

Fig. 6. Results from the ProVerif(2).

ProVerif can also be used to verify the perfect forward secrecy of the protocol by leaking some parameters. So, we employed ProVerif tool to implement our proposed authentication scheme and the authentication phase of the EV_i and the AGT_j are shown in Fig. 4.

Fig. 5 indicates the results from the Proverif. From Fig. 5, results (1) and (2) show that the adversary cannot obtain EV_i's private key k_i and AGT_j's private key k_j. Results (3) and (4) demonstrate the secrecy of the session keys SK_{ij} and SK_{ji}. Results (5)-(6) prove the anonymity of the EV_i and the AGT_j. Results (7) and (8) are the results of two injective correspondence assertions that show that the mutual authentication between EV_i and the AGT_j is valid. In addition, injectivity allows the EV_i and the AGT_j to check the freshness of their received messages which can resist replay attacks. Therefore, results (1)–(8) prove that our proposed authentication scheme achieves mutual authentication, session key security, user anonymity and can resist replay attacks.

Moreover, we also conducted experiments to demonstrate that the proposed scheme realizes perfect forward secrecy. In our experiments, the private keys of the EV_i and the AGT_j are transmitted over the public channel c, which means long-term keys are leaked to the adversary. As results (1) and (2) of Fig. 6 shows that both "not attacker(ki[])" and "not attacker(kj[])" are false, which proves that the adversary has obtained the k_i and the k_j. However, the results of "not attacker (SKij[])" and "not attacker (SKji[])" are still true. It demonstrates that even if the k_i and the k_j are leaked, the session key $\mathrm{SK}_{ij}(\mathrm{SK}_{ji})$ cannot be compromised. Therefore, the perfect forward secrecy is achieved in our proposed scheme.

B. Discussion on Possible Attacks

In Section VI-A, we have adopted ProVerif to demonstrate that the proposed scheme realizes mutual authentication, perfect forward secrecy, user anonymity, and session key security, and can resist replay attacks. So, in this section, we focus on some other attacks that we have not discussed in detail, such as smartcard stolen attacks, etc.

Authorized licensed use limited to: China University of Geosciences Wuhan Campus. Downloaded on September 06,2023 at 05:52:07 UTC from IEEE Xplore. Restrictions apply.

TABLE III
SECURITY FEATURES COMPARISONS OF RELATED SCHEMES

Security attributes	[3]	[21]	[22]	[23]	[24]	Ours
Replay attacks resistance	√	√	√	√	√	√
Man-in-the-middle attacks resistance	√	√	×	√	√	√
Modification attacks resistance	√	√	√	√	√	√
Privileged-Insider attacks resistance	√	√	×	√	-	√
Insider impersonation attacks resistance	√	√	×	√	√	√
Offline password guessing attacks resistance	√	√	-	-	-	√
Perfect forward secrecy	√	√	√	√	√	√
Session key security	√	√	√	√	√	√
Anonymity	√	√	√	×	√	√
Low computational cost	√	×	×	×	×	√
Low communication cost	√	√	×	×	×	√
Formal security analysis/proof	√	√	√	√	√	√
Automatic formal verification of security	×	×	×	×	×	√

Man-in-the-Middle Attacks: In our scheme, electric vehicle EV_i and aggregator AGT_j can share a session key $\mathrm{SK}_{ij}(\mathrm{SK}_{ji})$ only after they authenticate each other. In Section VI-A we have demonstrated that our scheme provides mutual authentication via ProVerif, so adversary A cannot establish independent connections with either AGT_j, or EV_i. Therefore, the adversary A cannot launch man-in-the-middle attacks successfully.

Modification Attacks: Suppose that an adversary A modifies message $\{C_3, \mathrm{Auth}_s\}$ with $\{C_3^*, \mathrm{Auth}_s^*\}$ and delivers this fraud message to electric vehicle EV_i to impersonate aggregator AGT_j. However, without the knowledge of the EV_i's identity ID_i and the AGT_j's secret key k_j, the adversary A cannot generate a valid A_i' to pass EV_i's verification. So, these attacks will be found when EV_i checks whether the equation $\mathrm{Auth}_s' = \mathrm{Auth}_s$ holds.

Furthermore, assume that an adversary A modifies message $\{C_1^*, C_2^*, M_i^*\}$ and then sends it to the AGT_j. Similarly, if A attempts to pass the AGT_j's verification, he/she requires to calculate a appropriate $C_2 = h(\mathrm{ID}_i||\mathrm{RID}_j||A_i||C_1)$. However, the adversary A cannot construct a valid A_i to make the equation $C_2^* = C_2$ equal. Therefore, our scheme can resist these attacks successfully.

Privileged-Insider Attacks: Assume that an adversary A is a privileged-insider user, and he/she possesses registration message $\{\mathrm{RID}_i, \mathrm{RPW}_i, \mathrm{ID}_i\}$ of the EV_i. Since the EVO's PW_i is protected by a secure hash function in our design, the adversary A cannot get the EVO's real identity ID_i and password PW_i, and thus cannot figure out the $I_0 = T_{\mathrm{ID}_i}(T_{h(\mathrm{ID}_i \oplus \mathrm{PW}_i)}(\mathrm{pub}_{\mathrm{TA}}))$ further. Therefore, we conclude that our scheme provides resistance of privileged-insider attacks.

Insider Impersonation Attacks: Assume a legal electric vehicle EV_a becomes a malicious adversary and try to impersonate another legal EV_i. However, without knowing the EV_i's ID_i, PW_i and Z, the EV_a cannot figure out the EV_i's private information A_i to pass the verification of the AGT_j. So, malicious electric vehicle EV_a cannot impersonate a legal electric vehicle EV_i to communicate with the AGT_j. On the other hand, when a registered aggregator AGT_b becomes a malicious attacker and try to impersonate another legitimate aggregator AGT_j, he/she needs to obtain AGT_j's private key k_j. However, without the knowledge of AGT_j's ID_j, r_j and s_j, the AGT_b cannot calculate k_j correctly. Therefore, our scheme achieves resistance of insider impersonation attacks.

Offline Password Guessing Attacks With/Without Smart Cards: Assume that an adversary A obtains all the messages relayed between the EV_i and the AGT_j and tries to launch offline dictionary attacks to get EV_i's password. To obtain the PW_i, the adversary A first needs to extract A_i from C_2. Even if the adversary A gets the k_i, he/she still cannot obtain PW_i without the knowledge of EV_i's ID_i, r_i and s_i. Therefore, the adversary A cannot launch offline dictionary attacks without smartcards successfully.

Suppose that an adversary A compromises all private information $\{\mathrm{RID}_{\mathrm{TA}}, I, k_i, Z, M_i, \mathrm{RID}_j, Q_j\}$ stored in the smartcard of the EV_i and performs offline dictionary attacks with smartcards. Compared with the offline dictionary attack without smartcards, the additional information known by the adversary A in these attacks is the information $\{\mathrm{RID}_{\mathrm{TA}}, I, k_i, Z, M_i, \mathrm{RID}_j, Q_j\}$ stored in the smartcard. According to the above discussion, the adversary A cannot obtain the EV_i's PW_i by using k_i. Furthermore, when the adversary A tries to extract PW_i from $I = T_{\mathrm{ID}_i}(T_{h(\mathrm{ID}_i \oplus \mathrm{PW}_i)k}(x))$, he/she will face the CMBDLP. Even if the adversary A solves the CMBDLP, without knowing the EV_i's ID_i and the TA's private key k_i, he/she still cannot guess PW_i correctly. Thus, the proposed scheme achieves resistance of offline password guessing attacks with/without smartcards.

VII. PERFORMANCE ANALYSIS

In this section, we evaluate the security features, computational costs, and communication costs of our scheme and those of five other competing schemes [1], [19]–[22].

A. Comparison of Security Features

The security features of our scheme and the other five related schemes [1], [19]–[22] are discussed in this section. As shown in Table III, Mahmood *et al.*'s authentication scheme [21] cannot achieve user anonymity. Odelu *et al.*'s authentication scheme [20] is vulnerable to impersonation attacks and man-in-the-middle attacks. Although Wazid *et al.*'s authentication scheme [19] and Kumar *et al.*'s scheme [22] are successful against common attacks, they involve time-consuming operations. Moreover, the related schemes [1], [19]–[22] do not provide automatic formal verification of security. According to Table III, our proposed scheme achieves resistance to known attacks and satisfies more security requirements in comparison with the other five related schemes [1], [19]–[22].

B. Computational Cost

In this section, the computational costs of our scheme and the other five related schemes [1], [19]–[22] are compared. In our experiments, we adopt OpenSSL library [40], GMP library [41], and PBC Library [42] to simulate these schemes on two Ubuntu 16.04 virtual machines with an Intel Pentium CPU G850 2.90-GHz processor, 4 GB of RAM. In

Authorized licensed use limited to: China University of Geosciences Wuhan Campus. Downloaded on September 06,2023 at 05:52:07 UTC from IEEE Xplore. Restrictions apply.

TABLE IV
EXECUTION TIME OF CRYPTOGRAPHIC ELEMENTS

Operations	Execution time on Intel Pentium G850
SHA1 (20 Bytes)	20385 ns
AES-256 encryption (16 Bytes)	6425 ns
AES-256 decryption (16 Bytes)	8513 ns
Bilinear Pairing (128 Bytes)	2361447 ns
EC point multiplication (160 bits)	803817 ns
EC point addition (160 bits)	16829 ns
Chebyshev Polynomial (160 bits)	237291 ns
Chebyshev Polynomial (256 bits)	379685 ns

TABLE V
COMPUTATIONAL COSTS COMPARISON

	$EV/MD_i/U_i/SM$	$AGT/SM_i/U_j/NAN$	Total
[3]	$7T_h+4T_c$ ≈1.403ms	$7T_h+4T_c+2T_s$ ≈1.526ms	$14T_h+8T_c+2T_s$ ≈2.929ms
[21]	$4T_m+2T_a+5T_h$ ≈3.764ms	$2T_m+8T_h$ ≈2.163ms	$6T_m+2T_a+13T_h$ ≈5.927ms
[22]	$2T_m+7T_h+2T_e+2T_b$ ≈9.774ms	$2T_m+7T_h+3T_e$ ≈6.515ms	$4T_m+14T_h+5T_e+2T_b$ ≈16.289ms
[23]	$4T_m+3T_a+4T_h$ ≈3.749ms	$3T_m+3T_a+4T_h$ ≈2.932ms	$7T_m+6T_a+8T_h$ ≈6.681ms
[24]	$3T_m+2T_s+6T_h$ ≈2.926ms	$3T_m+2T_s+7T_h$ ≈2.984ms	$6T_m+4T_s+13T_h$ ≈5.91ms
Ours	$5T_c+7T_h$ ≈1.717ms	$2T_c+5T_h$ ≈0.752ms	$7T_c+12T_h$ ≈2.469ms

our experiment, we select an elliptic curve over a finite field as $E_p(a,b)$: $y^2 = x^3 + ax + b(\mathrm{mod}\ p)$, where $a, b \in Z_p$, $4a^3 + 27b^2 \neq 0$ and p is a large prime. Specifically, the elliptic curve secp160r1 (SECG curve over a 160 bits prime field) from OpenSSL library was adopted in our simulation. We chose SHA1 as one-way hash function and 256-bit AES as symmetric key encryption/decryption operations in our experiments. In addition, the bit length of the modular operation is 1024 bits and the Chebyshev polynomial operation is 256 bits.

We first simulated different cryptographic elements with OpenSSL Library, PBC library, and GMP Library, including SHA1(20 Bytes), AES-256 encryption/decryption (16 Bytes), bilinear pairing (128 Bytes), EC point multiplication/addition (160 bits), and Chebyshev polynomial (160 and 256 bits). Each algorithm was performed 100 times and the average results are shown in Table IV. From Table IV, we summarize that Chebyshev polynomial operations are more efficient than the other operations presented in Table IV.

Then we carried out some simulations of our scheme and other five related schemes [1], [19]–[22]. The simulation results are illustrated in Table V. The symbol T_b, T_e, T_h, T_m, T_a, T_c, and T_s denote the time for executing a bilinear pairing operation, a modular exponentiation operation, a one-way hash function operation, a point multiplication operation of an elliptic curve, a point addition operation of an elliptic curve, a Chebyshev polynomial operation, and a symmetric key encryption/decryption operation, respectively.

From Table V, the Odelu *et al.*'s authentication scheme [20] requires performing four-point multiplication operations of

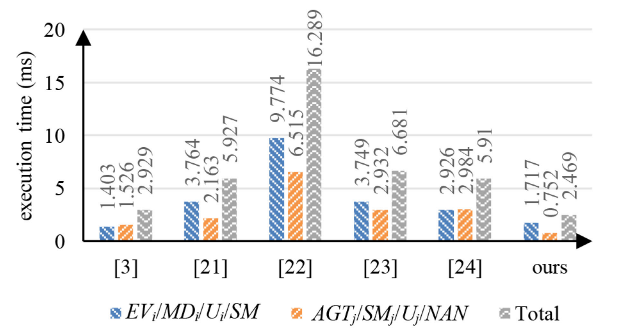


Fig. 7. Comparison of execution time in the login and authentication phase.

elliptic curve, fourteen hash operations, five modular exponentiation operations and two bilinear pairing operations to complete the authentication. Then, the execution time is given by $4T_m + 14T_h + 5T_e + 2T_b$ and the actual simulation time was 16.289 ms. From Table V, the computational costs of Odelu *et al.*'s authentication scheme are much higher than other related schemes [1], [19], [21], [22] and our scheme. That because Odelu *et al.*'s scheme [20] involves heavy-weight operations—bilinear pairing operations. In addition, from Table V, the total execution time of Wazid *et al.*'s authentication scheme [19], Mahmood *et al.*'s authentication scheme [21], and Kumar *et al.*'s authentication scheme [22] is 5.927, 6.681, and 5.91 ms respectively. Compared with Odelu *et al.*'s authentication scheme [20], these schemes [19], [21], [22] reduce the computational costs effectively by avoiding the use of bilinear pairing operations.

Furthermore, the proposed scheme requires to perform five Chebyshev polynomial operations and seven hash operations on the EV side, and needs to execute two Chebyshev polynomial operations and five hash operations on the AGT side. So, the total execution time is $7T_c+12T_h$ and the actual simulation time 2.469 ms. As shown in Table V, the computational costs of our authentication scheme and Abbasinezhad-Mood *et al.*'s authentication scheme [1] are 2.469 and 2.929 ms, which are much lower than other related schemes [19]–[22]. According to Table V, our authentication scheme and Abbasinezhad-Mood *et al.*'s authentication scheme [1] outperform the related schemes [19]–[22] in terms of the computational overhead. That is because efficient Chebyshev polynomial operations are employed in our authentication scheme and Abbasinezhad-Mood *et al.*'s authentication scheme [1].

As shown in Fig. 7, our authentication scheme achieves the best performance, taking only 2.469 ms in total. And on the AGT side, our scheme also achieves the lowest computational costs, which only takes 0.752 ms. Compared with other related schemes [1], [19]–[22], our authentication scheme reduces the computational costs up to 15.7%, 58.3%, 84.8%, 63.0%, and 58.2%, respectively. Therefore, our proposed authentication and key negotiation scheme is an energy-efficient authentication scheme and is suitable for SG environments.

C. Communication Cost

The comparison of communication costs between our scheme and other five related works [1], [19]–[22] is shown in Table VI. In our experiment, the user's ID is 64 bits, the output

Authorized licensed use limited to: China University of Geosciences Wuhan Campus. Downloaded on September 06,2023 at 05:52:07 UTC from IEEE Xplore. Restrictions apply.

TABLE VI
COMMUNICATION COSTS COMPARISON

Schemes	[3]	[21]	[22]	[23]	[24]	Ours
Cost(bits)	1376	1536	1920	2112	2240	1312

TABLE VII
STORAGE COSTS COMPARISON

	$EV/MD_i/SM$	$AGT/SM_i/NAN$
[3]	$1280n$	$1152m$
[21]	$1600n$	$640m$
[22]	$3328n$	$3328m$
[24]	$1664n$	$1376m$
Ours	$1312n$	$736m$

of hash function is 20 bytes (160 bits), the output of modular exponentiation operation is 32 bytes (256 bits), the output of Chebyshev polynomial is 32 bytes (256 bits), the output of an ECC operation is 40 bytes (320 bits), and the output of a timestamp is 4 bytes (32 bits). In addition, for bilinear pairing, the elements in group G_1 and G_2 are 128 bytes (1024 bits) and 128 bytes (1024 bits).

As shown in Table VI, our authentication scheme achieves the smallest communication load which is 1312 bits. And the communication costs for other related schemes [1], [19]–[22] are 1376, 1536, 1920, 2112, and 2240 bits, respectively. Obviously, the proposed scheme reduces communication costs in comparison with the related schemes [1], [19]–[22].

D. Storage Cost

In this section, we discuss the storage cost by comparing our proposed scheme with other four related schemes [1], [19], [20], [22]. Since the communication entities are both users in Mahmood *et al.*'s scheme [21] which is different from the other four schemes [1], [19], [20], [22] and our scheme, the storage cost comparison does not include Mahmood *et al.*'s scheme [21]. In our design, the EV needs to store the secure information $\{\mathrm{RID_{TA}}, I, k_i, Z, M_i, \mathrm{RID}_j, Q_j\}$ in its smartcard, where $\mathrm{RID_{TA}}$, k_i, Z, M_i, and RID_j are 160 bits, respectively and I, Q_j are 256 bits, respectively. Therefore, the storage cost required at the EV side is $1312\times n$ bits in our scheme. Here n denotes the number of AGT that the EV can communicate with. Furthermore, in our scheme, the AGT needs to store information $\{\mathrm{RID_{TA}}, \mathrm{RID}_j, Q_j, k_j\}$, where $\mathrm{RID_{TA}}$, RID_j, and k_j are 160 bits, respectively, and Q_j is 256 bits. So, the storage cost required at the AGT side for m EV is $736\times m$ bits in our design. The storage costs of the other four related schemes are shown in Table VII. As illustrated in Table VII, our proposed scheme achieves the lowest storage cost at the AGT side in comparison with other four related schemes [1], [19], [20], [22], In addition, compared with scheme [19], [20], [22], our scheme also reduced the storage cost at the EV side.

VIII. CONCLUSION

In this article, we presented a novel Chebyshev polynomial algorithm by adopting a square matrix-based binary exponentiation algorithm. The proposed algorithm solved the security challenge of Chebyshev polynomial algorithm in practical application and realized efficient and secure Chebyshev polynomial computation. Then, we further designed a fast authentication scheme by employing the proposed algorithm for SG environments. Since only lightweight Chebyshev polynomials and hash functions are used during the authentication and key negotiation processes, our design reduces the computational and communication costs in comparison with the state-of-the-art authentication schemes. We also adopted ProVerif tool to prove the security of our proposed authentication scheme. The security analysis demonstrated that our proposed authentication scheme can defend against various attacks. Therefore, our proposed authentication scheme is suitable for SG environments due to tackling both security and performance.

REFERENCES

[1] D. Abbasinezhad-Mood, A. Ostad-Sharif, S. M. Mazinani, and M. Nikooghadam, "Provably-secure escrow-less Chebyshev chaotic map-based key agreement protocol for vehicle to grid connections with privacy protection," *IEEE Trans. Ind. Informat.*, vol. 16, no. 12, pp. 7287–7294, Dec. 2020.
[2] P. Gope and B. Sikdar, "An efficient privacy-preserving authentication scheme for energy Internet-based vehicle-to-grid communication," *IEEE Trans. Smart Grid*, vol. 10, no. 6, pp. 6607–6618, Nov. 2019.
[3] Z. Wan, N. Xiong, N. Ghani, A. V. Vasilakos, and L. Zhou "Adaptive unequal protection for wireless video transmission over IEEE 802.11e networks," *Multimedia Tools Appl.*, vol. 72, no. 1, pp. 541–571, 2014.
[4] C. Lin, Y.-X. He, and N. Xiong, "An energy-efficient dynamic power management in wireless sensor networks," in *Proc. 5th Int. Symp. Parallel Distrib. Comput.*, 2006, pp. 148–154.
[5] M. Qi and J. Chen, "Two-pass privacy preserving authenticated key agreement scheme for smart grid," *IEEE Syst. J.*, early access, May 21, 2020, doi: 10.1109/JSYST.2020.2991174.
[6] A. Irshad, M. Usman, S. A. Chaudhry, H. Naqvi, and M. Shafiq, "A provably secure and efficient authenticated key agreement scheme for energy Internet-based vehicle-to-grid technology framework," *IEEE Trans. Ind. Appl.*, vol. 56, no. 4, pp. 4425–4435, Jul./Aug. 2020.
[7] S. A. Chaudhry, A. Albeshri, N. Xiong, C. Lee, and T. Shon, "A privacy preserving authentication scheme for roaming in ubiquitous networks," *Clust. Comput.*, vol. 20, no. 2, pp. 1–14, Mar. 2017.
[8] L. Kelly, A. Rowe, and P. Wild, "Analyzing the impacts of plug-in electric vehicles on distribution networks in British Columbia," in *Proc. Electr. Power Energy Conf.*, 2010, pp. 1–6.
[9] L. Xiong, N. Xiong, C. Wang, X. Yu, and M. Shuai, "An efficient lightweight authentication scheme with adaptive resilience of asynchronization attacks for wireless sensor networks," *IEEE Trans. Syst., Man, Cybern., Syst.*, early access, Dec. 20, 2019, doi: 10.1109/TSMC.2019.2957175.
[10] K. Clement-Nyns, E. Haesen, and J. Driesen, "The impact of charging plug-in hybrid electric vehicles on a residential distribution grid," *IEEE Trans. Power Syst.*, vol. 25, no. 1, pp. 371–380, Feb. 2010.
[11] G. Xu *et al.*, "A security-enhanced certificateless aggregate signature authentication protocol for InVANETs," *IEEE Netw.*, vol. 34, no. 2, pp. 22–29, Mar./Apr. 2020.
[12] S. Deilami, A. S. Masoum, P. S. Moses, and M. A. S. Masoum, "Real-time coordination of plug-in electric vehicle charging in smart grids to minimize power losses and improve voltage profile," *IEEE Trans. Smart Grid*, vol. 2, no. 3, pp. 456–467, Sep. 2011.
[13] Y. Su, G. Shen, and M. Zhang, "A novel privacy-preserving authentication scheme for V2G networks," *IEEE Syst. J.*, vol. 14, no. 2, pp. 1963–1971, Jun. 2020.
[14] P. Bergamo, P. D'Arco, A. De Santis, and L. Kocarev, "Security of public-key cryptosystems based on Chebyshev polynomials," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 52, no. 7, pp. 1382–1393, Jul. 2005.
[15] L. Kocarev, J. Makraduli, and P. Amato, "Public-key encryption based on Chebyshev polynomials," *Circuits Syst. Signal Process.*, vol. 24, pp. 497–517, Oct. 2005.

Authorized licensed use limited to: China University of Geosciences Wuhan Campus. Downloaded on September 06,2023 at 05:52:07 UTC from IEEE Xplore. Restrictions apply.

[16] F. Chen, X. Liao, T. Xiang, and H. Zheng, "Security analysis of the public key algorithm based on Chebyshev polynomials over the integer ring Z_N," *Inf. Sci.*, vol. 181, no. 22, pp. 5110–5118, 2011.
[17] D. Xiao, X. Liao, and S. Deng, "A novel key agreement protocol based on chaotic maps," *Inf. Sci.*, vol. 177, no. 4, pp. 1136–1142, 2007.
[18] X. Wang and J. Zhao, "An improved key agreement protocol based on chaos," *Commun. Nonlinear Sci. Numer. Simulat.*, vol. 15, no. 12, pp. 4052–4057, 2010.
[19] M. Wazid, A. K. Das, N. Kumar, and J. J. P. C. Rodrigues, "Secure three-factor user authentication scheme for renewable-energy-based smart grid environment," *IEEE Trans. Ind. Informat.*, vol. 13, no. 6, pp. 3144–3153, Dec. 2017.
[20] V. Odelu, A. K. Das, M. Wazid, and M. Conti, "Provably secure authenticated key agreement scheme for smart grid," *IEEE Trans. Smart Grid*, vol. 9, no. 3, pp. 1900–1910, May 2018.
[21] K. Mahmood, S. A. Chaudhry, H. Naqvi, S. Kumari, X. Li, and A. K. Sangaiah, "An elliptic curve cryptography based lightweight authentication scheme for smart grid communication," *Future Gener. Comput. Syst.*, vol. 81, pp. 557–565, Apr. 2018.
[22] P. Kumar, A. Gurtov, M. Sain, A. Martin, and P. H. Ha, "Lightweight authentication and key agreement for smart metering in smart energy networks," *IEEE Trans. Smart Grid*, vol. 10, no. 4, pp. 4349–4359, Jul. 2019.
[23] D. Wu and C. Zhou, "Fault-tolerant and scalable key management for smart grid," *IEEE Trans. Smart Grid*, vol. 2, no. 2, pp. 375–381, Jun. 2011.
[24] J. Xia and Y. Wang, "Secure key distribution for the smart grid," *IEEE Trans. Smart Grid*, vol. 3, no. 3, pp. 1437–1443, Sep. 2012.
[25] J.-L. Tsai and N.-W. Lo, "Secure anonymous key distribution scheme for smart grid," *IEEE Trans. Smart Grid*, vol. 7, no. 2, pp. 906–914, Mar. 2016.
[26] D. He, L. Wang, H. Wang, and M. K. Khan, "Lightweight anonymous key distribution scheme for smart grid using elliptic curve cryptography," *IET Commun.*, vol. 10, no. 14, pp. 1795–1802, 2016.
[27] D. Abbasinezhad-Mood and M. Nikooghadam, "An anonymous ECC-based self-certified key distribution scheme for the smart grid," *IEEE Trans. Ind. Electron.*, vol. 65, no. 10, pp. 7996–8004, Oct. 2018.
[28] S. Chatterjee, S. Roy, A. K. Das, S. Chattopadhyay, N. Kumar, and A. V. Vasilakos, "Secure biometric-based authentication scheme using Chebyshev chaotic map for multi-server environment," *IEEE Trans. Depend. Secure Comput.*, vol. 15, no. 5, pp. 824–839, Sep./Oct. 2018.
[29] H. Wang, D. Guo, Q. Wen, and H. Zhang, "Chaotic map-based authentication protocol for multiple servers architecture," *IEEE Access*, vol. 7, pp. 161340–161349, 2019.
[30] D. Abbasinezhad-Mood and M. Nikooghadam, "Efficient anonymous password-authenticated key exchange protocol to read isolated smart meters by utilization of extended Chebyshev chaotic maps," *IEEE Trans. Ind. Informat.*, vol. 14, no. 11, pp. 4815–4828, Nov. 2018.
[31] L. Zhang, H. Luo, L. Zhao, and Y. Zhang, "Privacy protection for point-of-care using chaotic maps-based authentication and key agreement," *J. Med. Syst.*, vol. 42, no. 12, p. 250, 2018.
[32] L. Kocarev and Z. Tasev, "Public-key encryption based on Chebyshev maps," in *Proc. Int. Symp. Circuits Syst.*, 2003, pp. 28–31.
[33] L. Kocarev, "Chaos-based cryptography: A brief overview," *IEEE Circuits Syst. Mag.*, vol. 1, no. 3, pp. 6–21, 3rd Quart., 2001.
[34] M. S. Farash and M. A. Attari, "An efficient and provably secure three-party password-based authenticated key exchange protocol based on Chebyshev chaotic maps," *Nonlinear Dyn.*, vol. 77, nos. 1–2, pp. 399–411, 2014.
[35] T.-F. Lee, "Provably secure anonymous single-sign-on authentication mechanisms using extended Chebyshev chaotic maps for distributed computer networks," *IEEE Syst. J.*, vol. 12, no. 2, pp. 1499–1505, Jun. 2018.
[36] L. Zhang, "Cryptanalysis of the public key encryption based on multiple chaotic systems," *Chaos Solitons Fractals*, vol. 37, no. 3, pp. 669–674, 2008.
[37] H. F. Zhu, Y. F. Zhang, Y. Xia, and H. Y. Li, "Password-authenticated key exchange scheme using chaotic maps towards a new architecture in standard model," *Int. J. Netw. Security*, vol. 18, no. 2, pp. 326–334, 2016.
[38] Islam and S. K. Hafizul, "Provably secure dynamic identity-based three-factor password authentication scheme using extended chaotic maps," *Nonlinear Dyn.*, vol. 78, no. 3, pp. 2261–2276, 2014.
[39] K. B. Bruno Blanchet. *ProVerif: Cryptographic Protocol Verifier in the Formal Model*. Accessed: Mar. 1, 2021. [Online]. Available: http://prosecco.gforge.inria.fr/personal/bblanche/proverif/
[40] (2018). *OpenSSL*. [Online]. Available: https://www.openssl.org/
[41] (2016). *GMP*. [Online]. Available: https://gmplib.org/
[42] (2019). *PBC Library*. [Online]. Available: https://crypto.stanford.edu/pbc/

Liping Zhang received the Ph.D. degree in information security from Huazhong University of Science and Technology, Wuhan, China, in 2009.

She is an Associate Professor of Information and Network Security with China University of Geosciences, Wuhan. She has published over 30 research papers, most of which are refereed international journal papers, including IEEE/ACM/IET journal papers. Her research interests include network security, key management and distribution, and privacy protection.

Dr. Zhang is the Principal Grant Holder of three externally funded research projects.

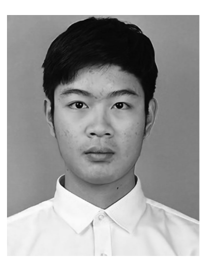
Yue Zhu received the B.Sc. degree in information security from China University of Mining and Technology, Xuzhou, China, in 2019. She is currently pursuing the Master degree in computer technology with China University of Geosciences, Wuhan, China.

Her research interests include communications security and network security.

Wei Ren (Member, IEEE) received the Ph.D. degree in computer science from Huazhong University of Science and Technology, Wuhan, China, in 2006.

He is a Full Professor with the School of Computer Science, China University of Geosciences, Wuhan, China, since 2013. He was with the Department of Electrical and Computer Engineering, Illinois Institute of Technology, Chicago, IL, USA, in 2007 and 2008, the School of Computer Science, University of Nevada Las Vegas, Las Vegas, NV, USA, in 2006 and 2007, and the Department of Computer Science, The Hong Kong University of Science and Technology, Hong Kong, in 2004 and 2005. He has published over 100 refereed papers, one monograph, and four textbooks.

Prof. Ren has obtained twenty patents and five innovation awards. He is a Distinguished Member of the China Computer Federation.



Yinghan Wang is currently pursuing the B.Sc. degree with the department of computer science, China University of Geosciences, Wuhan, China.

He conducted research on network security and privacy protection in medical systems as an undergraduate student in Zhang Liping's Lab. His research interests include privacy, authentication in wireless environment, and communication system security.

Authorized licensed use limited to: China University of Geosciences Wuhan Campus. Downloaded on September 06,2023 at 05:52:07 UTC from IEEE Xplore. Restrictions apply.

Kim-Kwang Raymond Choo (Senior Member, IEEE) received the Ph.D. degree in information security from Queensland University of Technology, Brisbane, QLD, Australia, in 2006.

He currently holds the Cloud Technology Endowed Professorship with The University of Texas at San Antonio (UTSA), San Antonio, TX, USA. He also has a courtesy appointment with the University of South Australia, Adelaide, SA, Australia.

Dr. K.-K. R. Choo is the recipient of the 2019 IEEE Technical Committee on Scalable Computing Award for Excellence in Scalable Computing (Middle Career Researcher), the 2018 UTSA College of Business Col. Jean Piccione and Lt. Col. Philip Piccione Endowed Research Award for Tenured Faculty, the Outstanding Associate Editor of 2018 for IEEE ACCESS, the British Computer Society's 2019 Wilkes Award Runner-Up, the 2014 Highly Commended Award by the Australia New Zealand Policing Advisory Agency, the Fulbright Scholarship in 2009, the 2008 Australia Day Achievement Medallion, and the British Computer Society's Wilkes Award in 2008. He has also received best paper awards from the IEEE SYSTEMS JOURNAL in 2021, *IEEE Consumer Electronics Magazine* for 2020, *EURASIP Journal on Wireless Communications and Networking* in 2019, IEEE TrustCom 2018, and ESORICS 2015; the Korea Information Processing Society's *Journal of Information Processing Systems* Outstanding Research Award (Most-cited Paper) for 2020 and Survey Paper Award (Gold) in 2019; the IEEE Blockchain 2019 Outstanding Paper Award; and Best Student Paper Awards from Inscrypt 2019 and ACISP 2005. He and his team won the Digital Forensics Research Challenge organized by Germany's University of Erlangen–Nuremberg in 2015. He is an ACM Distinguished Speaker and an IEEE Computer Society Distinguished Visitor (2021–2023), and included in Web of Science's Highly Cited Researcher in the field of Cross-Field–2020.

Neal N. Xiong (Senior Member, IEEE) received the Ph.D. degree in sensor system engineering from Wuhan University, Wuhan, China, in 2007, and the Ph.D. degree in dependable communication networks from Japan Advanced Institute of Science and Technology, Nomi, Japan, in 2008.

He is currently an Associate Professor (fifth year) with the Department of Mathematics and Computer Science, Northeastern State University, Tahlequah, OK, USA. Before he attended Northeastern State University, he worked with Georgia State University, Atlanta, GA, USA; Wentworth Technology Institution, Boston, MA, USA; and Colorado Technical University, Colorado Springs, CO, USA (Full Professor about five years) about ten years. He has published over 200 international journal papers and over 100 international conference papers. Some of his works were published in IEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS, IEEE or ACM transactions, ACM Sigcomm workshop, IEEE INFOCOM, International Conference on Distributed Computing Systems, and International Parallel and Distributed Processing Symposium. His research interests include cloud computing, security and dependability, parallel and distributed computing, networks, and optimization theory.

Dr. Xiong received the Best Paper Award in the tenth IEEE International Conference on High Performance Computing and Communications (2008) and the Best student Paper Award in the 28th North American Fuzzy Information Processing Society Annual Conference (2009). He is the Chair of "Trusted Cloud Computing" Task Force and IEEE Computational Intelligence Society. He has been a General Chair, Program Chair, Publicity Chair, Program Committee Member and Organizing Committee Member for over 100 international conferences, and as a Reviewer of about 100 international journals, including IEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS, IEEE Systems, Man, and Cybernetics Society (Park: A/B/C), IEEE TRANSACTIONS ON COMMUNICATIONS, IEEE TRANSACTIONS ON MOBILE COMPUTING, IEEE TRANSACTIONS ON PARALLEL AND DISTRIBUTED SYSTEMS. He serves as an Editor-in-Chief, Associate Editor, or Editor member for over ten international journals (including an Associate Editor for IEEE TRANSACTIONS ON SYSTEMS, MAN AND CYBERNETICS: SYSTEMS, ASSOCIATE EDITOR FOR INFORMATION SCIENCE, an Editor-in-Chief for *Journal of Internet Technology*, and an Editor-in-Chief for *Journal of Parallel and Cloud Computing*), and a Guest Editor for over ten international journals, including *Sensor Journal*, *Wireless Networks*, and *Mobile Networks and Applications*. He is a Senior Member of IEEE Computer Society from 2012.

Authorized licensed use limited to: China University of Geosciences Wuhan Campus. Downloaded on September 06,2023 at 05:52:07 UTC from IEEE Xplore. Restrictions apply.