I. J. Computer Network and Information Security, 2020, 3, 44-52
Published Online June 2020 in MECS (http://www.mecs-press.org/)
DOI: 10.5815/ijcnis.2020.03.05

Privacy Protection in Smart Cities by a Personal Data Management Protocol in Blockchain

Hossein Mohammadinejad
SheikhBahaee University, Isfahan, 8143153784, Iran
E-mail: mohammadi.n@shbu.ac.ir

Fateme Mohammadhoseini
SheikhBahaee University, Isfahan, 8143153784, Iran
E-mail: f.mohamadhoseini@gmail.com

Received: 11 December 2019; Accepted: 20 December 2019; Published: 08 June 2020

Abstract—Due to the increase of cybercrime and security risks in computer networks as well as violations of user privacy, it is essential to upgrade the existing protection models and provide practical solutions to meet these challenges. An example of these risks is the presence of a third party between users and various services, which leads to the collection and control of large amounts of users' personal information and the possibility of their databases being misused or hacked. Blockchain technology and encrypted currencies have so far shown that a decentralized network of peer-to-peer users, along with a general ledger, can do reliable computing. So, in this article, we are going to introduce a protocol that converts the blockchain network to an automated access control manager without the presence of a third party. To this end, we designed a mutual authentication protocol to create a secure channel between the user and the service and then demonstrate its accuracy and completeness using the Gong-Nidham-Yahalom belief logic [1]. The results of our evaluations show that our proposed protocol is secure enough to be used on the blockchain network and attackers are unable to penetrate, track, impersonate, inject, misrepresent or distort information using the common attacks.

Index Terms—Blockchain, Privacy, Secure Channel, Decentralized Personal Data Management, Authentication Protocol, Access Control.

I. Introduction

Today, information technology has made dramatic changes in the way people live. It manages societies and affects all the traits of communities by providing them with the knowledge and intelligence they need in society. Therefore, to provide the necessary framework for smart city development and information technology development, there is a need for a secure and reliable communication network for the transmission of data, covering its applications from simple activities to national and international discourses. It should be noted that establishing security in communications networks is a very complicated and expensive process, and its serious challenge is the dynamics and the up-to-dates. Until now, various types of security systems have been designed and developed, each with a specific aspect of security [2]. Given the inherent challenges of wireless communication and its insecure nature, as well as the issues of heterogeneity, the implementation of security mechanisms in various technologies is associated with its complexities. On the other hand, due to energy constraints on mobile users' devices, it is better to use relatively lightweight security mechanisms [3]. As the governments need to adapt to the future conditions of smart cities and e-commerce, a new approach in this research is the introduction of a decentralized personal data management system and the establishment of a secure communication channel in the blockchain network, which, in addition to security, is useful for large-scale deployment. This system, where the users themselves control the data without intermediaries, can significantly reduce the risks. Transactions in this system are defined to perform instructions such as data sharing, storage, and querying. This system requires a secure channel between the user and the server, which can be created using mutual authentication protocols. The main issue of our research is the creation of this secure channel.Our proposed protocol is a complement to the research presented by Zyskind et al. [33] They have assumed the safe channel in their article and we designed and evaluate this channel.

Ensuring the security of these protocols requires various evaluations and integrity of the structure and the step-by-step implementation of the protocol. So far, several models have been introduced to analyze the security of encryption protocols and algorithms, which can be summarized as follows:

1. Formal model
2. Computational model
3. Information theory model
4. System theory model
5. Unconditional or provable model

In a formal model, it is assumed that the algorithms or cryptographic primitives used in the protocols are secure or ideal. In this model, which uses both verification and Prove theorem methods, an attacker only has access to the communication channel and can eavesdrop or manipulate information from the communication channel. Indeed, the primary purpose of this model is to examine the structure of the security protocol and to validate its performance [4].

In the computational model, the basis of the evaluation is based on the assumption that an attacker with limited equipment and facilities will not be able to obtain useful information from encrypted messages in a reasonable time. Protocols or algorithms that have this feature are called computationally secure [5].

The basis of evaluation in the information theory model is the Mutual information between the original message and the encrypted message. Qualitatively, if the attacker, despite its unlimited computational power, fails to obtain any information on the probability distribution of possible messages in the active attacks, the protocol is secure from an information theory perspective [6].

In the system theory model, the main focus is on the well-known and common attacks. In this way, the evaluation of a protocol is made so that its resistance to these attacks is evident. In this view, a protocol is called practically secure, if it is not broken by any known common attacks in a reasonable time [7].

In the unconditional or provable model, all attacks against the protocol will also be considered, but in this view, a protocol has provable secure, whenever a Low-threshold for the average calculations required in any attack against it is provable, and this proof is not based on any unproven assumptions. This model is also known as the unconditional security model [6].

II. Related Work

The record of research on authentication protocols, which has been conducted for the last three decades, illustrates their application in different contexts. Due to the recent use of blockchain networking, there has been limited research on authentication protocols in this context. But since these protocols can be applied in different platforms with consideration of the conditions and limitations [8], in this article, we have studied a variety of famous and lightweight protocols and achieve our proposed protocol design criteria.

For the first time, human authentications are expressed by Hooper and Balm [9]. The HB protocol is based on the difficulty of solving the LPN[1] problem, and it is more Lightweight than all of its newer versions but is not robust against the man-in-the-middle and the GRS and active attacks. The confidentiality and integrity of the information are not guaranteed in this protocol [10]. Jules and Wise [11] illustrated that the HB protocol is only resistant to passive attacks, and they introduced a modified version of it called HB+. In the following, the AUHB++ protocol is introduced by Lee et al. [12] to correct the weaknesses of the improved HB+ protocol, namely, HB++. although it is resistant to the man-in-the-middle attack, it does not have privacy and has high computational cost too. The PUFHB protocol is a tamper-resistant protocol of the HB family that was developed by Hemmori et al. [13] with the aim to modify the HB+ protocol. But in addition to not being resistant to any man-in-the-middle, GRS, and spoofing attacks, RFID tags also need additional hardware to run it. Monila and Pinado [14], introduced the HB-MP protocol and introduced a new way to exchange information efficiently, but Long et al. [15] presented a man-in-the-middle attack that effectively compromised the HB-MP protocol. They also developed the HB-MP protocol and presented the HB-MP+ protocol to resist the attack. Gilbert et al. [16] proposed two new RANDOM-HB # and HB# protocols. RANDOM-HB# imposed an unacceptable cost on labels, but HB# increased the efficiency of the RANDOM-HB # protocol. Khaled Awafi et al. [17] presented a general man-in-the-middle attack against the two protocols mentioned. Tian et al. [18] presented a new lightweight authentication protocol called UAPP. UAPP makes little use of computing resources (storage and communication), but asynchronous attacks can put the security of this protocol at serious risk. The HB-MP + protocol is resistant to a man-in-the-middle attack, but authentication is done in several iterations that are time-consuming. Although the HB# protocol completes the task in one iteration and does not use a hash function, it is vulnerable to a man-in-the-middle attack. Yun et al. [19] presented the HB-MP ++ protocol with ultra-lightweight functions, but it is not robust against GRS and spoofing attacks, also not capable of mutual authentication and has a high computational cost. The NL-HB protocol was developed by Makandan et al. [20], the NL-HB protocol was developed by Makandan et al. [13], which achieved the same security with shorter key lengths than the HB protocol, but despite having a low complexity cost, against any man-in-the-middle, GRS and spoofing attacks are not resistant. Samia et al. [21] introduced the RC-HB protocol as a new version of the HB protocol that uses shorter key lengths and has lower communication costs. However, this protocol is not resistant to any man-in-the-middle, GRS, and spoofing attacks. Zhicai et al. [22] presented the IHB protocol as a modified version of the HB+ protocol. This protocol is generally better than other versions of the HB family of protocols. But it is not resistant to GRS attacks and spoofing. Khoureich [23] has also proposed another protocol called LhHB that is more lightweight to his hHB protocol, which is more practical to implement than the previous version. The protocol, despite its advantages over other HB versions, is not resistant to GRS attacks and spoofing. The TREAD protocol [24] eliminates the drawbacks of classical methods, but it should be noted that this feature is based on the use of a cryptographic algorithm and a digital signature method and has high computational overhead.

[1] Learning from Parity with Noise

Pagnin et al. [25] developed a lightweight hybrid authentication protocol called HB+DB, in which the HB+ protocol was combined with the idea of distance constraint. It is worth noting that the GRS attack against HB+DB breaks down and stops authentication. Kiltz et al. [26] developed protocols for authentication and message authentication (MAC) codes whose security is based on the LPN problem and is resistant to an active man-in-the-middle attack. Jeong et al. [27] also proposed a multi-agent authentication method for MCC in their research. Dey et al. [28] present their authentication protocol using a message summary called MDA. Omri et al. [29] proposed a way for the user to use the handwriting as an authentication factor in cloud access. Schwab and Yang [30] present their proposed authentication protocol, FDZ, for mobile device validation in the cloud computing environment. Finally, it should be noted that the performance of mobile devices has several different limitations, as Abolfazli et al. [31] have noted in their paper. The idea of remote computing and using mobile devices by cloud-based computing and storage resources to overcome the inherent challenges and weaknesses in mobile computing has attracted the attention of researchers to provide a wider range of services required by users [32].

III. Proposed Method

Our proposed protocol is a complement to the research presented by Zyskind et al. [33] In this study, a secure channel for user and service communication was not created; they have assumed this channel. Therefore, in this research, we provide the secure channel required for the mentioned system.

III.1 Secure channel formation

A. The preliminary phase

In this phase, the various security parameters to build a secure channel and authentication and key agreement, are calculated.

1) First is selected the elliptic curve equation $E_p(a,b)$: $y^2 = x^3 + ax + b \pmod p$, on a field F_P and a base point P, also called the generating point [34], on $E_p(a, b)$ With a one-way hash function $h(\cdot)$: $\{0,1\}^* \rightarrow \{0,1\}^k$.

2) U stores each service's ID_i in an *ID* table. Next, U also assigns a *UID* identifier to himself.

3) U selects a random integer $sec \in Z_p^*$ as the secret key for use in the symmetric code system and generates a random integer $sk < n$ as a private key where n is the base-order rank or *P* generator. It then calculates the public key corresponding to this private key, $pk = skP$.

4) The public / private key pair (*pk*, *sk*) is used for the asymmetric password signature system. In this step, u calculates $C_1 = E_{sec}(ID_i)$ and $C_2 = UIDP$ for each service s. Sec and *sk* are kept secret by U. Also, u sends the public key *pk* and the secret pair (C_1, C_2) to S.

B. Authentication and establishment phase of the secure communication channel

During the authentication process, U and S perform the following four steps for authentication and key agreement.

1) S Selects a random integer $r_1 \in_R Z_p^*$ to compute $C_3 = e_{pk}(ID_i \| C_1 \| r_1)$ in which $e_{pk}(\bullet)$ represents the public-key encryption function using *pk* and C1 (that *pk* is belonging to U, and $C_1 = E_{sec}(ID_i)$ is the secret value of S). Subsequently, S sends $C_3 = e_{pk}(ID_i \| C_1 \| r_1)$ to U.

2) In this step, U obtains the values of ID_i, C_1, and r_1 by decoding the C_3 message using the private key *sk*. It then checks its validity by matching the ID_i to the *ID* table. If it is not valid, the authentication process will stop. Otherwise, it decodes C_1 using the *sec*, (the secret key), to obtain the ID_i. It then compares the ID_i value in C_3 with the same value obtained from C_1 decoding. If these values are not the same, it terminates the authentication process; otherwise, U selects two random integers $r_2 \in Z_p^*$ and $r_3 \in Z_p^*$ to compute $SK = h(r_1 \| r_2)$, (that SK is shared session key), and calculates the $C_4 = E_{r1}(UID \| r_2)$ authentication message. Where $E_{r1}(\bullet)$ is the symmetric encryption algorithm using the secret key r_1. Finally, U sends the message (C_4, r_3) to S.

It should be noted; there is no need to encode a random integer r_3, since it is only used to check the message's newness and has nothing to do with the final session key. Even if an attacker has a random integer r_3, there is no risk of disclosing the shared key. Therefore, the random integer r_3 is sent explicitly, and this method is widely used in various protocols to check the novelty of the message.

1) S After receiving the message (C_4, r_3), using r_1, decodes the encrypted sequence C_4 and obtains the values of r_2 and *UID*. Then, by computing *UIDP*, it checks for the equation C_2=*UIDP*. If C_2 is equal to *UIDP*, calculates the joint session key $SK' = h(r_1 \| r_2)$ and the authentication message $C_5 = h(SK' \| (r_3 + 1))$ and sends C_5 to U. Otherwise, S will reject the message and terminates the authentication process.

2) U After receiving the C_5 message, checks for the value of C_5 equal to the calculated value $h(SK \| (r_3 + 1))$. If these two values are the same, it will consider *SK* as the session key with S; otherwise, it terminates the authentication process.

In the proposed protocol, the session key is constructed by two random integers with high entropy freely chosen by U and S. It should be noted that the session key in each authentication process and key agreement will be different, that is, the secret pair (C_1, C_2) is not related to the final computed session key. Therefore, even if the confidential pair (C_1, C_2) is compromised, information about the session key is not disclosed, and the attacker cannot obtain the exchanged messages between U and S that encrypted by the session key.

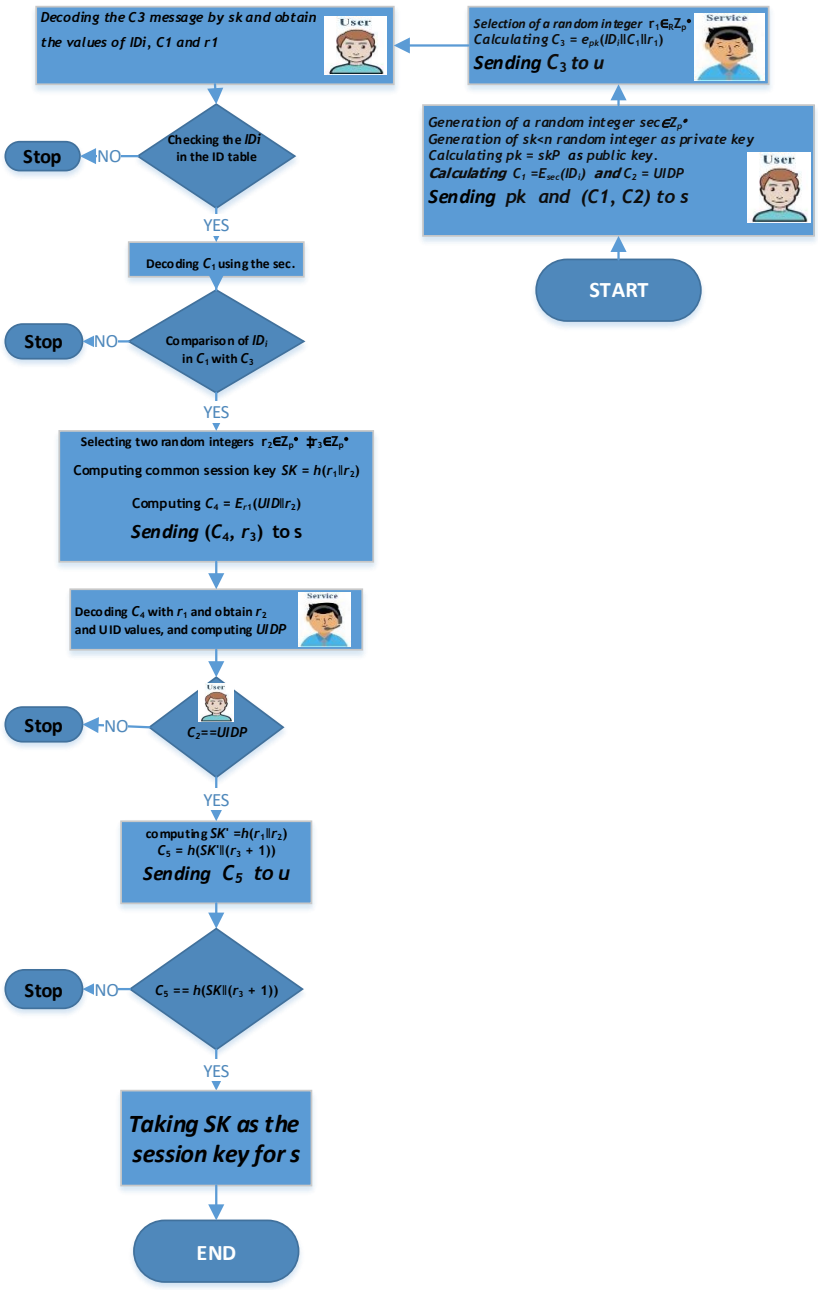

Fig.1. flowchart of the protocol steps

IV. Describing the Objectives of the Protocol for Formal Evaluation

In this section, some symbols are modified to fit the GNY logic, so the protocol is converted to P→Q:(X). Also, the user's private key is represented by –K, and the corresponding public key is +K.

$$s \to u: \left(\left\{ ID_i \,\|\, \{ID_i\}_{sec} \,\|\, r_1 \right\}_{+K} \right) \tag{1}$$

$$u \to s: \left(\left\{ UID \,\|\, r_2 \right\}_{r1}, r_3 \right) \tag{2}$$

$$s \to u: \left(h\left(h\left(r_1 \,\|\, r_2 \right) \,\|\, \left(r_3 + 1 \right) \right) \right) \tag{3}$$

In the following, we describe the aims according to GNY logic to prove the validity of this protocol, which includes three different aspects.

A. Message authentication

Objective 1: U believes that the message is identifiable in the first step.

$$u \mid\equiv \phi\left\{ ID_i \,\|\, \{ID_i\}_{sec} \,\|\, r_1 \right\}_{+K} \tag{4}$$

Objective 2: S believes that the message can be identifiable in the second step.

$$s \mid\equiv \phi\left(\left\{UID \parallel r_2\right\}_{r1}, r_3\right) \tag{5}$$

Objective 3: U Believes that the message can be identifiable in the third step.

$$u \mid\equiv \phi\left(h\left(h\left(r_1 \parallel r_2\right) \parallel \left(r_3+1\right)\right)\right) \tag{6}$$

B. Sender authentication

Objective 4: S believes that in the second step, U has sent the message.

$$s \mid\equiv u \mid \sim \left\{UID \parallel r_2\right\}_{r1} \tag{7}$$

Objective 5: U believes that in the third step, S sends the message.

$$u \mid\equiv s \mid \sim h(h(r_1 \parallel r_2) \parallel (r_3+1)) \tag{8}$$

C. Establishment of the session key's preliminary data

Objective 6: S Believes that U believes that SK is a shared key between S and U.

$$s \mid\equiv u \mid \equiv s \xleftrightarrow{SK} u \tag{9}$$

Objective 7: S Believes that SK is a shared key between S and U.

$$s \mid\equiv s \xleftrightarrow{SK} u \tag{10}$$

Objective 8: U believes that S is SK's owner.

$$u \mid\equiv s \ni SK \tag{11}$$

Objective 9: U believes that S believes that SK is a shared key between S and U.

$$u \mid\equiv s \mid\equiv s \xleftrightarrow{SK} u \tag{12}$$

V. The Assumptions Considered in the Proposed Protocol

Assumptions using GNY logic, with mention reasoning and using GNY language are listed below.

1. U generates the secret key sec in this protocol, so sec belongs to U. U also is the owner of the private key - K and the public key + K.

$$u \ni sec, u \ni +K, u \ni -K \tag{13}$$

2. Since U holds the ID table, so it believes that ID_i is identifiable.

$$u \mid\equiv \phi\left(ID_i\right) \tag{14}$$

3. Since S holds the value of $C_2 = UIDP$ secretly and holds the base point P or the same generator, it can check the UID and believe that the UID is identifiable.

$$s \mid\equiv \phi\left(UID\right) \tag{15}$$

4. In our protocol, the random integer r_1 is generated by S, so S is the owner of r_1 and believes that r_1 is new.

$$s \ni r_1, s \mid\equiv \#\left(r_1\right) \tag{16}$$

5. The random integer r_1 is generated by S as part of the temporary session key, only for the current session. So we assume that S believes r_1 is a suitable shared key for itself and U.

$$s \mid\equiv s \xleftrightarrow{r_1} u \tag{17}$$

6. Random integers r_2 and r_3 are generated by U, so U owns r_2 and r_3 and believes that r_3 is identifiable and r_2 is new.

$$u \ni r_3, u \mid\equiv \phi\left(r_3\right), u \ni r_2, u \mid\equiv \#\left(r_2\right) \tag{18}$$

7. Since SK is a temporary session key for the current session generated by U, therefore, we assume that U believes SK is a suitable shared key between itself and S.

$$u \mid\equiv u \xleftrightarrow{SK} s \tag{19}$$

8. S believes that U is capable of producing SK as a primary string to generates the appropriate session key between U and S.

$$s \mid\equiv u \mid \Rightarrow s \xleftrightarrow{SK} u \tag{20}$$

VI. Proof of Authentication Using GNY Logic

In this section, we analyze the protocol using GNY logic. We present only the rational principles needed to prove the validity of the proposed protocol and the stated objectives:

A. First connection

$$\frac{u \mid\equiv \phi\left(IDi\right),\ u \ni sec}{u \mid\equiv \phi\left\{ID_i\right\}_{sec},\ u \mid\equiv \phi(ID_i \parallel \left\{ID_i\right\}_{sec} \parallel r_1)} \tag{21}$$

If U believes that ID_i is identifiable and U owns the sec key, then U will believe that ID_i encryption is identifiable with the sec key, so the formula $\{ID_i \| \{ID_i\}_{sec} \| r_1\}$ is also identifiable.

$$\frac{u \mid\equiv \phi\left(ID_i \mathbin{/\!/} \{ID_i\}_{sec} \mathbin{/\!/} r_1\right),\ u \ni +K}{u \mid\equiv \phi\{ID_i \mathbin{/\!/} \{ID_i\}_{sec} \mathbin{/\!/} r_1\}_{+K}} \tag{22}$$

If U believes that ($ID_i \| \{ID_i\}_{sec} \| r_1$) is identifiable and U owns the public key +K, then it believes that encoding $\{ID_i \| \{ID_i\}_{sec} \| r_1\}_{+K}$ is identifiable. Therefore, in this protocol, U can detect the message $\{ID_i \| \{ID_i\}_{sec} \| r_1\}_{+K}$ in the first run. (Objective 1)

B. Second connection

$$\frac{s \mid\equiv \phi(UID),\ s \ni r_1}{s \mid\equiv \phi(UID \| r_2),\ s \mid\equiv \phi\{UID \| r2\}_{r1}} \tag{23}$$

If S believes that *UID* is identifiable, then S will believe that the formula ($UID \| r_2$), where *UID* is a component of it, is identifiable. Since S owns r_1, he also believes that $\{UID \| r_2\}_{r1}$ encoding is identifiable.

$$\frac{s \mid\equiv \phi\{UID \| r_2\}_{r1}}{s \mid\equiv \phi(\{UID \| r_2\}_{r1}, r_3)} \tag{24}$$

If S believes that $\{UID \| r_2\}_{r1}$ is identifiable, then it will believe that ($\{UID \| r_2\}_{r1}$, r_3) that $\{UID \| r_2\}_{r1}$ is a component of it, is also identifiable. Therefore, it can be concluded that in the proposed protocol, S can detect the message ($\{UID \| r_2\}_{r1}$, r_3) in the second communication. (Objective 2)

$$\frac{s \lhd^* \{UID \| r_2\}_{r1},\ s \ni r_1,\ s \left|\equiv s \xleftrightarrow{r_1} u,\ s\right| \equiv \phi(UID \| r_2),\ s \mid\equiv\ \#(r_1)}{s \mid\equiv u \mid\sim \{UID \| r_2\}_{r1},\ s \mid\equiv u \ni r_1} \tag{25}$$

If the following five conditions are met:

1) *S gets the formula* ($UID \| r_2$) that *is encrypted with the* r_1 *key and marked "not originated here".*
2) *S owns* r_1.
3) *S believes that* r_1 *is a shared key appropriate to himself and U.*
4) *S believes that the formula* ($UID \| r_2$) is *identifiable.*
5) *S believes that* r_1 *is new.*

In this case, S will believe that:

1) *U once has sent the message* ($UID \| r_2$) that *encrypted with* r_1.
2) *U holds* r_1. *(Objective 4)*

Now according to the GNY logic, we assume that $s \mid\equiv u \mid\Rightarrow u \mid\equiv^*$, *meaning S believes that U is truthful and competent, so we can deduce the following statement:*

$$\frac{s \mid\equiv u \mid\Rightarrow u \mid\equiv^*, s \mid\equiv u \mid\sim (\{UID \| r_2\}_{r1}, r_3) \sim> u \mid\equiv s \xleftrightarrow{SK} u), s \mid\equiv \#(\{UID \| r_2\}_{r1}, r_3)}{s \mid\equiv u \mid\equiv s \xleftrightarrow{SK} u} \tag{26}$$

If S believes that U is honest and competent and S receives the following message:

$$(\{UID \| r_2\}_{r1},\ r_3) \sim> u \mid\equiv s \xleftrightarrow{SK} u) \tag{27}$$

That believes U sent it.

Then S must have believed that U believes in $u \left|\equiv s \xleftrightarrow{SK} u\right.$. Therefore, S believes that U believes that SK is a suitable shared key between S and U. (objective 6)

$$\frac{s \mid\equiv u \mid\Rightarrow s \xleftrightarrow{SK} u,\ s \left|\equiv\ u\right| \equiv\ s \xleftrightarrow{SK} u}{s \mid\equiv s \xleftrightarrow{SK} u} \tag{28}$$

If S believes that U is a competent authority in the expression of $s \xleftrightarrow{SK} u$ and U believe $s \xleftrightarrow{SK} u$, then S must believe $s \xleftrightarrow{SK} u$. Therefore, S believes that SK is an appropriate shared key between itself and U. (objective 7)

C. Third Communication

$$\frac{u \lhd \left\{ID_i \mathbin{/\!/} \{ID_i\}_{sec} \mathbin{/\!/} r_1\right\}_{+K},\ u \ni -K}{u \lhd (ID_i \mathbin{/\!/} \{ID_i\}_{sec} \| r_1), u \lhd r_1} \tag{29}$$

If U is told that the formula ($ID_i \| \{ID_i\}_{sec} \| r_1$) is encrypted with the public key +*K* and he also owns the corresponding private key -*K*, then it is considered that he knows the decrypted content of that formula. Also, r_1 was mentioned as a component of the formula.

$$\frac{u \lhd r_1, u \ni r_2, u \ni r_3}{u \ni r_1, u \ni (r_1 \| r_2), u \ni h(r_1 \| r_2), u \ni (r_3 + 1)} \tag{30}$$

If r_1 is told to U, he can have r_1, and if U owns r_2, then he can have ($r_1 \| r_2$) and $h(r_1 \| r_2)$. For this reason, if U has r_3, then U will also have (r_3+ 1).

$$\frac{u \ni h(r_1 \| r_2),\ u \ni (r_3 + 1)}{u \ni (h(r_1 \| r_2) \| (r_3 + 1))} \tag{31}$$

If U owns $h(r_1 \| r_2)$ and (r_3+1), then ($h(r_1 \| r_2) \mathbin{/\!/} (r_3 + 1)$) also holds.

$$\frac{u \mid\equiv \phi(r_3)}{u \mid\equiv \phi(h(r_1 \| r_2) \| (r_3 + 1))} \tag{32}$$

If U believes that r3 is identifiable, then U believes that (r3 + 1) is identifiable, and (h (r1‖r2) ‖ (r3 + 1) that (r3 + 1) is a component of it, will be identifiable too.

$$\frac{u \mid\equiv \phi(h(r_1 \| r_2) \| (r_3 + 1)),\ u \ni (h(r_1 \| r_2) \| (r_3 + 1))}{u \mid\equiv \phi(h(r_1 \| r_2) \| (r_3 + 1))} \tag{33}$$

If U believes that ($h(r_1 \| r_2) \mathbin{/\!/} (r_3 + 1)$) is identifiable and owns ($h(r_1 \| r_2) \mathbin{/\!/} (r_3 + 1)$), then it will believe Where $h(h(r_1 \| r_2) \mathbin{/\!/} (r_3 + 1))$ is identifiable. Thus, U can be believed that the message $h(h(r_1 \| r_2) \mathbin{/\!/} (r_3 + 1))$ is identifiable in the third communication. (Objective 3).

$$\frac{u \mid\equiv \#(r_2),\ u \ni (r_1 \| r_2)}{u \mid\equiv \#(r_1 \| r_2),\ u \mid\equiv \#(h(r_1 \| r_2))} \tag{34}$$

If u believes that r2 is new, also owns (r1‖r2), then it will believe that (r1‖r2) as well as h (r1‖r2) are new.

$$\frac{u \triangleleft^{*} h((r_3+1),\ <SK>),\ u \ni ((r_3+1),\ SK),\ u \mid\equiv u \xleftrightarrow{SK} s,\ u \mid\equiv \#(SK)}{u \mid\equiv s \mid\sim ((r_3+1),\ <SK>),\ u \mid\equiv s \mid\sim h((r_3+1),\ <SK>)} \tag{35}$$

If the following four conditions are met:

- U gets a formula consisting of a one-way function of (r_3+1), and *SK* is marked "does not originate here."
- U owns (r_3+1) and *SK*.
- U believes that *SK* is a shared key appropriate to himself and S.
- U believes that *SK* is new.

In this case U will believes that S has sent once ((r_3+1), *SK*) and $h(h(r_1 \| r_2) \| (r_3+1))$.

Then we can say that U believes S sends the message $h(h(r_1 \| r_2) \| (r_3+1))$ in the third communication of the protocol. (Objective 5).

$$\frac{u \mid\equiv s \mid\sim ((r_3+1), SK), u \mid\equiv \#(SK)}{u \mid\equiv s \mid\sim SK, u \mid\equiv s \ni SK} \tag{36}$$

If U believes that S has sent the formula ((r_3+1), *SK*) once, it will then be believed that S has sent *SK* once. Also, if U believes that *SK* is new, then it will believe that S owns *SK*. Therefore, U believes that SK is held by S. (Objective 8)

According to the GNY logic, we assume $u \mid\equiv s \mid\Rightarrow s \mid\equiv^{*}$, meaning U believes that S is truthful, so we can deduce the following statement:

$$\frac{u \mid\equiv s \mid\Rightarrow s \mid\equiv^{*}, u \mid\equiv s \mid\sim (h(SK \| (r_3+1)) \sim> s \mid\equiv s \xleftrightarrow{SK} u), u \mid\equiv \#(SK \| (r_3+1)}{u \mid\equiv s \mid\equiv s \xleftrightarrow{SK} u} \tag{37}$$

If U believes S to be honest and truthful and receives the following message,

$$h(SK \| (r_3+1)) \sim> s \mid\equiv s \xleftrightarrow{SK} u \tag{38}$$

That believes is sent by S, then U must believe that S believes $s \xleftrightarrow{SK} u$. Therefore, we can conclude in the proposed protocol, U believes SK is a suitable shared key between S and itself. (Objective 9)

As we have seen, all nine objectives have been achieved to prove the assumed authentication protocol, and we reached our desired conclusion.

As this protocol does not duplicate any response between the service and the user, the attacker cannot use a tracing attack against this protocol. That is, the random value of r1 generated by the services to calculate C3 and the r2 and r3 values generated by the user to calculate and construct SK, C4, and (C4, r3) messages for each session, makes all responses are non-repetitive and easily block the tracking attack.

Reasons of robustness of our proposed protocol are listed below:

Deal with repeat attack can easily be done by using the session key in any communication, and we have done so in the proposed protocol (we create a new SK for each connection), meaning, that the SK session key is updated before each session is established and the attacker cannot make the repeat attack on the messages being sent. In this process, the random integer r_3 is also used to check the novelty of the message while having nothing to do with the final session key (r_3 is nonce).

Spoofing Attacks on mutual authentication protocols, not applicable at all. As a result, the attacker cannot replace himself Instead of the user, and deceive the services.

Since all data is encrypted between the user and the services. And before each session, the SK key is generated on both sides to convey each message, there is no useful packet for the attacker to make a man-in-the-middle attack.If he intends to make the attack by breaking the encrypted packet, in the first step, the public key algorithm based on elliptic curves and must solve a discrete logarithm problem, and elliptic curve discrete logarithm is one of intractability and practically Insoluble problems.

Since in our proposed protocol, session keys generated during different sessions are independent of the public and private keys with long-term usage and also session keys used in previous sessions, there is the feature of perfect forward secrecy in the key exchange mechanism. It means the shared session key is made by performing the hash function to the accession of two random values r1 (created by services) and r2 (generated by user). That independent of the set of public and private keys, and by the randomness and reproduction of each session, it is also independent of the session keys used in previous sessions. As a result, an attacker cannot access confidential information in the future by storing the exchanged data on the communication channel due to increasing decryption power (for example, the use of computers and quantum computing to solve complex problems).

VII. Computational Complexity Analysis of Proposed Protocol

In this section, we first explain the features of our proposed protocol, then evaluate and calculate its computational cost. As mentioned earlier, this protocol provides a secure channel for communication between the user and the service, with the key agreement feature and mutual authentication.To evaluate the computational cost, we first introduce the following symbols:

Table 1. Symbols needed to evaluate the computational cost

T_m	Time to run a modular exponentiation operation
T_e	Time to perform an elliptic curve scalar multiplication operation .
T_h	Time to run a one-way hash function.
T_{se}	Time to run a symmetric-key encryption operation.
T_{sd}	Time to run a symmetric-key decoding operation
T_{ae}	Time to run an asymmetric-key encryption operation
T_{ad}	Time to run an asymmetric-key decoding operation

In our proposed protocol, the computational cost of the preliminary phase on the user side is $T_e + T_{se}$. In the authentication phase, the computing cost on the service side is $T_{ae} + 2T_h + T_{sd} + T_e$,and on the user side is $T_{ad} + T_{sd} + 2T_e + 2T_h + 2T_{se}$ Therefore, the total computational cost of our proposed protocol is $3T_e + T_{ae} + T_{ad} + 2T_{sd} + 2T_{se} + 4T_h$.

Theoretical analysis and experimental results show that the modular exponentiation operation T_m and T_{ae}/T_{ad} asymmetric encoding / decoding operation, have much higher computational cost than the T_{se}/T_{sd} symmetric encoding/decoding operation and T_e elliptic curve scalar multiplication operation. Also, compared to the T_{ae}/T_{ad} asymmetric encoding/decoding operation and the T_m modular exponentiation operation, the computational cost of the T_h hash function is so low that it can be ignored.

Since our proposed protocol avoids costly modular exponentiation operations and reduces the number of asymmetric encryption/ decryption operations, it is efficient. Also, compared to integer factorization based protocols, our protocol reduces the computational cost on the user side.

In this protocol, the user needs to store a hash function, ID and (C_1, C_2, *pk*, *P*) confidential information, that the length of each of the C_2 and *P* binary strings is 1024 bits, the length of each of the C_1 and ID strings are 32 bits, and *Pk* is 128 bits. As a result, the total amount of storage overhead on the user side is 2240 bits.

$$(1024 \times 2) + (32 \times 2) + 128 = 2240 \text{ Bit} \qquad (39)$$

Thus, we have shown that our proposed protocol is well-suited for use in blockchain platforms for privacy in smart cities, Because of the low computational cost on the user side, it can be implemented on the blockchain platform for types of users with different processing power.

VIII. Conclusion

In general, it is advisable not to share personal information and sensitive data with a third party as a trusted party, as there is a possibility of abuse or attacks against them. Therefore, with a major change of strategy, users must control their data without various security threats or limit the ability of companies and providers to provide specific services and own their data and information. Our proposed solution enables this possibility by establishing a secure channel on the platform of blockchain (intended as an access controller). In this case, users do not need to trust any third-parties and always are aware of the data collected about them and how they are used. At the same time, blockchain easily recognizes users as the owners of their personal information. Companies can also focus on better services using the data without worrying about the responsibility of users' data security. Also, with a decentralized platform, legal and regulatory decisions on the collection, storage, and sharing of sensitive information will be more comfortable. That is, laws and regulations can be defined within the blockchain and implemented automatically. In other cases, the general office may act as a legal document for accessing (or storing) data because it is computationally tamper-proof. In this regard, in order to establish a secure communication channel between the user and the server, we have introduced a mutual authentication protocol that has privacy and key agreement features. Due to this protocol's features, it is resistant to conventional attacks such as tracking, replay, spoofing, man-in-the-middle, injecting, or distorting information. And it has the leading privacy feature to resist future attacks.

References

[1] Gong, L., R. Needham, and R. Yahalom. Reasoning about belief in cryptographic protocols. in Proceedings. 1990 IEEE Computer Society Symposium on Research in Security and Privacy. 1990. IEEE.

[2] Jangirala, S., A.K. Das, and A.V. Vasilakos, Designing secure lightweight blockchain-enabled RFID-based authentication protocol for supply chains in 5G mobile edge computing environment, in IEEE Transactions on Industrial Informatics. 2019.

[3] Riesco, R., X. Larriva-Novo, and V. Villagra, Cybersecurity threat intelligence knowledge exchange based on blockchain. Telecommunication Systems, 2019: p. 1-30.

[4] Cremers, C., M. Dehnel-Wild, and K. Milner, Secure authentication in the grid: A formal analysis of DNP3 SAv5. Journal of Computer Security, 2019. 27(2): p. 203-232.

[5] Souri, A. and M. Norouzi, A state-of-the-art survey on formal verification of the internet of things applications. Journal of Service Science Research, 2019. 11(1): p. 47-67.

[6] Zhao, G., et al. Design and Formal Verification of a VANET Lightweight Authentication Protocol. in 2018 IEEE 18th International Conference on Communication Technology (ICCT). 2018. IEEE.

[7] Saxena, M. and A. Dua. Security solutions against attacks in mobile ad hoc networks and their verification using BAN logic. in 2017 International Conference on Energy, Communication, Data Analytics and Soft Computing (ICECDS). 2017. IEEE.

[8] Lee, J.H., Systematic approach to analyzing security and vulnerabilities of blockchain systems. 2019, Massachusetts Institute of Technology.

[9] Hopper, N.J. and M. Blum, A secure human-computer authentication scheme. 2000, CARNEGIE-MELLON UNIV PITTSBURGH PA SCHOOL OF COMPUTER SCIENCE.

[10] Karrothu, A., R. Scholar, and J. Norman. An analysis of LPN based HB protocols. in 2016 Eighth International Conference on Advanced Computing (ICoAC). 2017.

IEEE.
[11] Juels, A. and S.A. Weis. Authenticating pervasive devices with human protocols. in Annual international cryptology conference. 2005. Springer.
[12] He, L., et al. An Improved HB++ Protocol Against Man-in-Middle Attack in RFID System. in 2008 4th International Conference on Wireless Communications, Networking and Mobile Computing. 2008. IEEE.
[13] Hammouri, G. and B. Sunar. PUF-HB: A tamper-resilient HB based authentication protocol. in International Conference on Applied Cryptography and Network Security. 2008. Springer.
[14] Munilla, J. and A. Peinado, HB-MP: A further step in the HB-family of lightweight authentication protocols. Computer Networks, 2007. 51(9): p. 2262-2267.
[15] Leng, X., K. Mayes, and K. Markantonakis. HB-MP+ protocol: An improvement on the HB-MP protocol. in 2008 IEEE international conference on RFID. 2008. IEEE.
[16] Gilbert, H., M.J. Robshaw, and Y. Seurin. : Increasing the Security and Efficiency of. in Annual International Conference on the Theory and Applications of Cryptographic Techniques. 2008. Springer.
[17] Ouafi, K., R. Overbeck, and S. Vaudenay. On the security of HB# against a man-in-the-middle attack. in International Conference on the Theory and Application of Cryptology and Information Security. 2008. Springer.
[18] Tian, Y., G. Chen, and J. Li, A new ultralightweight RFID authentication protocol with permutation. IEEE Communications Letters, 2012. 16(5): p. 702-705.
[19] Yoon, B., et al. HB-MP++ protocol: An ultra light-weight authentication protocol for RFID system. in 2009 IEEE International Conference on RFID. 2009. IEEE.
[20] Madhavan, M., et al. NLHB: A light-weight, provably-secure variant of the HB protocol using simple non-linear functions. in 2010 National Conference On Communications (NCC). 2010. IEEE.
[21] Ali, S.A., R.M. Mohamed, and M.H. Fahim. RCHB: Light-weight, provably-secure variants of the HB protocol using rotation and complementation. in 2011 5th International Conference on Network and System Security. 2011. IEEE.
[22] Shi, Z., et al., An Improved HB+ Protocol and its Application to EPC Global Class-1 Gen-2 Tags. International Journal of Security and Its Applications, 2015. 9(8): p. 211-220.
[23] Khoureich, K.A., Light-hHB: A new version of hHB with improved session key exchange. Cryptology ePrint Archive, Report 2015/713, 2015.
[24] Avoine, G., et al. A terrorist-fraud resistant and extractor-free anonymous distance-bounding protocol. in Proceedings of the 2017 ACM on Asia Conference on Computer and Communications Security. 2017. ACM.
[25] Pagnin, E., et al., HB+ DB: Distance bounding meets human based authentication. Future Generation Computer Systems, 2018. 80: p. 627-639.
[26] Kiltz, E., et al. Efficient authentication from hard learning problems. in Annual International Conference on the Theory and Applications of Cryptographic Techniques. 2011. Springer.
[27] Jeong, Y.S., J.S. Park, and J.H. Park, An efficient authentication system of smart device using multi factors in mobile cloud service architecture. International Journal of Communication Systems, 2015. 28(4): p. 659-674.
[28] Dey, S., S. Sampalli, and Q. Ye. Message digest as authentication entity for mobile cloud computing. in 2013 IEEE 32nd International Performance Computing and Communications Conference (IPCCC). 2013. IEEE.
[29] Omri, F., et al. Cloud-ready biometric system for mobile security access. in International Conference on Networked Digital Technologies. 2012. Springer.
[30] Schwab, D. and L. Yang. Entity authentication in a mobile-cloud environment. in Proceedings of the Eighth Annual Cyber Security and Information Intelligence Research Workshop. 2013. ACM.
[31] Abolfazli, S., et al., Rich mobile applications: genesis, taxonomy, and open issues. Journal of Network and Computer Applications, 2014. 40: p. 345-362.
[32] Aminzadeh, N., Z. Sanaei, and S.H. Ab Hamid, Mobile storage augmentation in mobile cloud computing: Taxonomy, approaches, and open issues. Simulation Modelling Practice and Theory, 2015. 50: p. 96-108.
[33] Zyskind, G. and O. Nathan. Decentralizing privacy: Using blockchain to protect personal data. in 2015 IEEE Security and Privacy Workshops. 2015. IEEE.
[34] Dahab, R. and J. López, An overview of elliptic curve cryptography. Institute of Computing State University of Campinas Brazil, Brazil, 2000.

Authors' Profiles

Hossein Mohammadinejad is an assistant professor of computer engineering at the Faculty of Engineering of the Sheikhbahaee University. He received his B.Sc. in computer engineering in 2000 from University of Tehran, Tehran, Iran, and his M.Sc. in computer engineering from University of Isfahan, Isfahan, Iran, in 2003. He received his Ph.D. in Computer Engineering in 2017 from University of Isfahan, Isfahan, Iran.

Fateme Mohammadhoseini received her B.Sc. degree in Software Engineering from University of Isfahan, Isfahan, Iran in 2005. She is now M.Sc. Student of Department of Computer Engineering at Sheikhbahaee University, Isfahan, Iran. Her research interests include reliability in Wireless Network and Smart Grid.

How to cite this paper: Hossein Mohammadinejad, Fateme Mohammadhoseini, "Privacy Protection in Smart Cities by a Personal Data Management Protocol in Blockchain", International Journal of Computer Network and Information Security(IJCNIS), Vol.12, No.3, pp.44-52, 2020. DOI: 10.5815/ijcnis.2020.03.05