The ISC Int'l Journal of Information Security

ISeCure

January 2022, Volume 14, Number 1 (pp. 57–68)

http://www.isecure-journal.org

A Privacy Preserving Mutual Authentication Scheme Suitable for IoT-Based Medical Systems

Mahdieh Ebrahimi [1], Majid Bayat [1,*], and Behnam Zahednejad [2]

[1] *Department of Computer Engineering, Shahed University, Tehran, Iran.*

[2] *Institute of Artificial Intelligence and Blockchain, Guangzhou University, Guangzhou, China.*

ARTICLE INFO.

Article history:
Received: May 1, 2019
Revised: January 19, 2021
Accepted: July 7, 2021
Published Online: August 28, 2021

Keywords:
Internet of Things, Medical System, Security, Authentication, Privacy Preserving, ProVerif

Type: Research Article

doi: 10.22042/ISECURE.2021.183936.463

dor: 20.1001.1.20082045.2022.14.1.5.5

ABSTRACT

The medical system remains among the fastest to adopt the internet of things. The reason for this trend is that integration internet of things (IoT) features into medical devices greatly improve the quality and effectiveness of service. However, there are many unsolved security problems. Due to medical information is critical and important, authentication between users and medical servers is an essential issue. Recently, Park *et al.* proposed an authentication scheme using Shamir's threshold technique for IoT-based medical information system and claimed that their scheme satisfies all security requirements and is immune to various types of attacks. However, in this paper, we show that Park *et al.*'s scheme does not achieve user anonymity, forward security, and mutual authentication and it is not resistant to the DoS attacks and then we introduce an improved mutual authentication scheme based on elliptic curve cryptography (ECC) and Shamir's secret sharing for IoT-based medical information system. In this paper, we formally analyze the security properties of our scheme via the ProVerif. Moreover, we compare our proposed scheme with other related schemes in terms of security and performance.

1 Introduction

The IoT has had a significant development recently. It actually aims at creating communication for every thing using the least amount of computational cost [1–3]. With the quick development of IoT and gaining the attention of researchers and companies, the traditional medical system moved to the IoT environment rapidly [4–6].The doctors can remotely cure their patients. A number of sensors are connected to the patients which captures the required information for their treatment and sends this information to the medical servers through wireless connections. This method is really efficient for those patients who have to be controlled and monitored constantly or those who have settled in for a way areas. Actually, IoT technology in medical environments facilitates the health system management and presents several more efficient services in this industry. Despite the fact that there are lots of advantages in using IoT in health industry, there are also plenty of security challenges in this field due to the sensitivity of medical information. One of the most important existing challenge would be the authentication between the users and medical servers [7, 8].

[*] Corresponding author.

Email addresses: mahdiyeh.ebrahimi@shahed.ac.ir, mbayat@shahed.ac.ir, Zahednejadb@gmail.com

ISSN: 2008-2045 © 2020 ISC. All rights reserved.

1.1 Related Work

The telecare medical information system (TMIS) is one of the systems which uses the provided services by communication technologies. In TMIS, the patients are able to receive various medical services through the Internet. Since the medical information are sensitive and the Internet is not secure, it is critical to make sure in terms of confidentiality, integrity and patients authentication. Numerous authentication schemes have been provided to make sure of the patient's secure access to the medical servers through internet [9–14]. In 2012, Chen *et al.* [11] introduce a dynamic-ID based authentication scheme for TMIS. In 2013, Cao *et al.* [12] showed that Chen *et al.*'s scheme [11] is not resistant to off-line identity guessing attack and un-detectable on-line password guessing attack. Then presented an improved scheme to confront with these attacks. Xie *et al.* also found several security vulnerabilities in Chen *et al.*'s scheme [11] and presented an improved version of this scheme. In 2015, Amin *et al.* [13] proposed a smart card based remote user authentication scheme via ECC. In 2016, chaudhry *et al.* [14] analyzed the Amin *et al.*'s scheme [13] and proved their scheme is not resistant to stolen smart card and stolen verifier attacks. Also this scheme is having scalability issues along with inefficient password change and password recovery phases. Then chaudhry *et al.* [14] proposed an authentication scheme using ECC and smart card for multi server TMIS architecture. Qiu *et al.* [15] analyzed the chaudhry *et al.* [14] scheme and developed a scheme in 2017 to meet the security shortcomings such as off-line password guessing attack, user/server impersonation attack, and man-in-middle attack. Ostad Sharif *et al.* [16] carried out a security analysis on the Qiu *et al.* scheme in 2018 and found out it neither had neither user anonymity nor resistance to a key compromise impersonation attack. They then introduced a new authentication and session key agreement scheme to cover the security gaps. IoT is a new technology that provides human-human, human-thing and thing-thing interactions. A huge amount of sensitive and personal information is exchanged in these interactions; therefore, security and privacy are challenges in IoT. Many authentication schemes have been presented to maintain the security in IoT [17–21]. Recently, using IoT in the medical industry has been resulted to create IoT-based medical systems. In 2014 Xu *et al.* [22] presented an IoT-based system for emergency medical services in which an overall process for IoT-based medical systems has been conceptually mentioned. In 2015, Hou *et al.* [23] proposed an authentication scheme for IoT-based healthcare systems and formally proved its security. In 2017, Park *et al.* [24] presented a scheme for authentication IoT-based medical systems. In this scheme, the users first choose their own things and then the server authenticates users to access to the group of things [20, 25]. In this paper, we show that the Park *et al.*'s scheme [24] suffers some vulnerabilities such as DoS, lack of anonymity, mutual authentication and forward security and we propose an improved mutual authentication scheme in IoT for medical systems based on elliptic curve cryptography(ECC) and Shamir's secret sharing.

1.2 Research Contributions

In this paper, we introduce a secure authentication scheme for IoT-based medical system, as follows:

(1) One of the most important security requirements in the internet of things is the mutual authentication. So in our proposed scheme, mutual authentication between RA (registration authority) and the user is considered.
(2) It is essential that the privacy and anonymity of users are provided in the authentication schemes for IoT. Therefore, our scheme satisfies users anonymity property.
(3) In our scheme the attacker cannot compute session keys of the past sessions even if the secret key k (secret key between server and RA) is revealed. So, in the proposed scheme forward security is considered.
(4) Our scheme is resistant to common attacks such as DoS attack, replay attack, user impersonation attack.
(5) We introduce the formal security analyze of our scheme with using the popular automated tool ProVerif.

1.3 Paper Organization

The rest of this paper is organized as follows. In Section 2, we bring some related preliminaries. We review Park *et al.*'s scheme [24] and its security weaknesses in Section 3. We present an improved scheme in Section 4. The security analysis and formal verification of the proposed scheme are introduced in Section 5. Finally, a conclusion is given in Section 6.

2 Preliminaries

In this section, we introduce some definitions of Shamir's secret sharing and elliptic curve cryptography that are used in our scheme. Then we describe symbols used in this paper.

2.1 Elliptic Curve Cryptography

The elliptic curve [26, 27] over Z_p, $p > 3$ is the set of all pairs $x, y \in Z_p$ which fulfill

$$y^2 \equiv x^3 + ax + b \ mod \ p \tag{1}$$

together with an imaginary point of infinity O, where $a, b \in Z_p$ and the condition

$$4a^3 + 27b^2 \neq \ 0 \ mod \ p. \tag{2}$$

Definition 1. Let be ε/F an elliptic curve and $P, Q \in \varepsilon(F)$ two points. Finding the smallest integer n such that $Q = nP$ is called the elliptic curve discrete logarithm problem (ECDLP). The value n is known as the discrete logarithm of Q.

2.2 Shamir's Secret Sharing

A (t,n) threshold secret sharing scheme is a technique for n parties to transport shares s_i of a message s such that any t of them to reconstruct the message, but so that not $t-1$ of them can easily do so. The threshold scheme is perfect if knowledge of $t-1$ or fewer shares provides no information regarding s. For more information on this scheme, readers can refer to the original paper [28].

2.3 Notations

We list the symbols throughout the paper in Table 1.

Table 1. Notations used in this paper

Notation	Description
ID , pw	identity/password of a user
s_i	ith session identity
t_i	identity of Thing i
p_i	public point (x_i, y_i) of Thing i on the arbitrary polynomial
r_i/R_i	random number generated by users/server
k	shared key between the server and the RA
t/n	specific things chosen by a user/total things
$f_u(x)$	polynomial generated by the server for authenticating a user
p_r	arbitrary point (x_r, y_r) on the polynomial $f_u(x)$
h_1	mapping Z_p to 0,1
h_2	mapping Z_p to (x_1, y_1) on the polynomial $f_u(x)$
MID, MPW	mask of identity/password of a user
T_i	time stamp
r_a/r_b	random number
P	elliptic curve generator

2.4 Security Requirements

We list the security requirements for an authentication scheme in IoT based medical systems as follows.

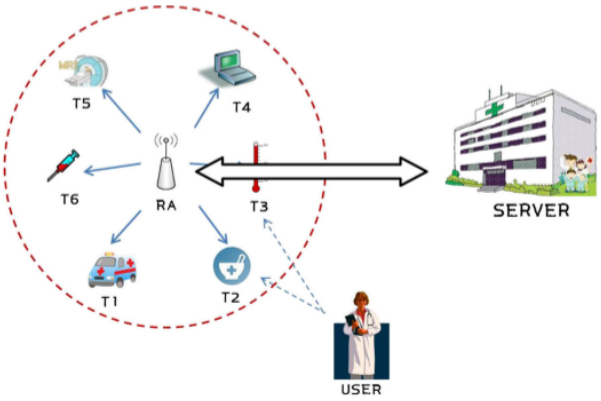

Figure 1. Overview of an IoT-based medical system

(1) Mutual authentication: Both the RA and the user must authenticate each other.
(2) Anonymity: The identity of the user should not be revealed by the attacker. The authentication protocol must ensure that the identity of a user is kept secret during the communication.
(3) Forward security: It means that the attacker cannot compute session keys of the past sessions even if the long term private keys of the protocol participants are revealed.
(4) Non-manipulation: No users can computationally manipulate an authentication value.
(5) Resistance to common attacks: Attacks such as the replay attack, user impersonation attacks, insider attacks, and denial-of-service attacks must be avoided.

3 Cryptanalysis of Park *et al.*'s Scheme

In this section, we review Park *et al.*'s scheme [24] and its security weaknesses.

3.1 Review of Park *et al.*'s Scheme

This scheme is an authentication scheme on the Internet of things. The process is as follows:

(1) First, the user chooses the things that he/she wants to communicate with them, and sends the same session id to all.
(2) The things forward requests after request reception from the user, to RA.
(3) The RA requests the server for identifying the user.
(4) Server checks the user and sends information for authenticating.
(5) The RA authenticates the user based on the information received from the server.

Figure 1 shows a process of authentication between a user and a thing in IoT-based medical environment.

Park *et al.*'s scheme consists of two phases: registration and user authentication. They are explained as follows:

3.1.1 Registration

User to server: In Park *et al.*'s scheme, the user selects identity and password, uid and upw, and random number $r_i \in Z_q^*$, then computes a temporary password $tpw = h_1(r_i \| upw)$. Finally, he/she sends (uid, tpw) to the server via a secure channel.

3.1.2 User Authentication

(1) User to t things: Access requests are sent to intended things with session id and user id by the user, (s_1, uid).

(2) t Things to RA: A request to the RA is sent by each thing to authenticate the user with session id, user id and its public point, $(s_1, uid, p_1), (s_1, uid, p_2), \ldots,\ (s_1, uid, , p_t)$.

(3) RA to server: First, user id and the information related to the things are saved by the RA. Then the requests came from the things are sent to server with session id, user id and public points of t things, $(s_1, uid, p_1, p_2, \ldots, p_t)$ by RA. Provided that the user is legitimate on the network and has access to things, this request is accepted by the server. In the end, server selects a random number R_i and computes polynomial $f_u(x)$ using points $(p_1, p_2, \ldots, p_t, h2(tpw|R_i))$.

(4) Server to user: Random number R_i and session id are sent to the user by server, (s_1, R_i).

(5) Server to RA: A random point p_r is selected from the polynomial $f_u(x)$, then this random point with session id is encrypted by shared k and sent to RA, $(s_1, E_k(s_1, p_r))$.

(6) User to RA: Polynomial $f_u(x)$ is computed by the user using public points $(p_1, p_2, \ldots, p_t)$, password upw and random number R_i. Then authentication value $AUTH = h_1(s_1, uid, f_u(0))$ is created by polynomial $f_u(x)$ and it is sent to RA with session id and user id, $(s_1, uid, AUTH)$.

(7) RA: The polynomial $f_u(x)$ is also computed by the RA using public points $(p_1, p_2, \ldots, p_t)$ and the random point p_r given by the server. Then an $AUTH = h_1(s_1, uid, f_u(0))$ is computed by the RA and compared with $AUTH$. If $AUTH = AUTH'$ is true, the user is accepted by RA. If not, the user is rejected by the RA.

3.2 Weaknesses of Park *et al.*'s Scheme

In this section, we find out that Park *et al.*'s scheme is vulnerable to denial of service attack and it does not satisfy forward key secrecy, mutual authentication and anonymity properties. The details of the proposed attacks are described as follows:

3.2.1 Lack of Anonymity

One of the important properties of authentication is the user anonymity. This property makes authentication mechanism more strong. Anonymity means that an attacker cannot determine which user interacts with things and RA. In this scheme, the user id is clearly transmitted on the public channel. Users send their access requests to the things, (s_i, uid), and each thing sends (s_i, uid, p_t) to RA on the public channel. As a result, the attacker can discover which user and thing are connected together by eavesdropping the messages on the channel. Therefore, Park *et al.*'s scheme fails to preserve user anonymity.

3.2.2 Lack of Mutual Authentication

One of the most significant security requirements on the IoT is mutual authentication. That means, the two sides of the communication must authenticate each other. In Park *et al.*'s scheme, the user can calculate $f_u(x)$ using the random number received by the server, R_i. Then the user calculates the value of $AUTH$ using $f_u(x)$ and sends $(s_i, uid, AUTH)$ to RA. RA obtains the value of P_r by decrypting the $(s_1, Ek(s_1, p_r))$ and calculates $AUTH'$ using that value then compares it with the $AUTH$, that the user sent to RA. If these two values are equal, the user is authenticated to the RA. As far as we concerned, in this scheme only the user is authenticated to the RA and the RA is not authenticated to the user, so this scheme has no mutual authentication.

3.2.3 Denial of Service Attack

In the denial of service attack, a malicious user can suspend the services of server by flooding it with fake messages such as login requests. In Park *et al.*'s scheme, users can send a lot of connection requests to things without any restrictions and the things also forward these requests to the RA. Consequently, the RA fails because of the large number of requests. Therefore, this scheme is not resistant to the DoS attack.

3.2.4 Lack of Forward Security

Forward secrecy means that the attacker cannot find session keys created in past sessions even if he/she discovers the private values of the network. Because if the attacker gets the previous session keys, he can decrypt previous exchanged messages and gain more information. In this scheme, assuming that the attacker can find the key between the server and RA, he can find p_r from $(s, Ek(s, p_r))$ and computes the polynomial $f_u(x)$ using public points $(p_1, p_2, \ldots, p_t)$ and p_r then he can calculate session key between the user and RA, $K_1 = h_1(f_u(0))$. So this scheme lacks forward secrecy.

4 The Proposed Scheme

In this section, to overcome the weaknesses of Park *et al.*'s scheme, we propose an improved authentication scheme for IoT based medical system. Our scheme is a mutual authentication in IoT for medical systems. Our scheme consists of four entities: server, RA, things and users. Users include medical staff and patients who want to use network services. All users go to the server to register on the network. We assume that the server has lists of users' ID/password pairs and eligibility of users to access things. Things are medical devices that serve users on the network. Things are assigned their public points p_i before distribution. In this scheme, things are clustered according to the purpose of use and relative proximity and for better control, a regional authority (RA) manages each cluster. The size of the cluster can be changed dynamically according to the efficiency, security, and purpose. The server shares a pairwise key k with the RA and the RA can communicate securely with the server using this pre-shared key. So a clustered IoT is composed of a RA and n things with which users can communicate.
The process of our scheme is as follows:

(1) First, the user chooses the things that he/she wants to communicate with them, and sends the same session id to all.
(2) The things forward requests after request received from the user, to RA.
(3) The RA collects all of the requests received from the things and forwards them to the server.
(4) The server checks whether the user is legitimate on the network and has the access to the desired things. If the user is legitimate, server sends the required information for the authentication to the user and RA.
(5) The RA and user authenticate each other based on the information received from the server.
(6) Finally, the selected things receive a signal from the RA regarding user authentication.

Figure 2 shows a process of authentication between a user and things in IoT-based medical environment. In our scheme, based on the Shamir's secret sharing algorithm, assumed that adversaries compromise no more than *(t-1)* out of n things in a given time period. Practically, it is hard to compromise t things which are designed securely. We also assume that the RA has more computation and communication power than things. More precisely, an RA has an additional powerful radio to establish wireless links with things and strong resistance against malicious attacks.
Our scheme includes two phases, registration and authentication as follows:

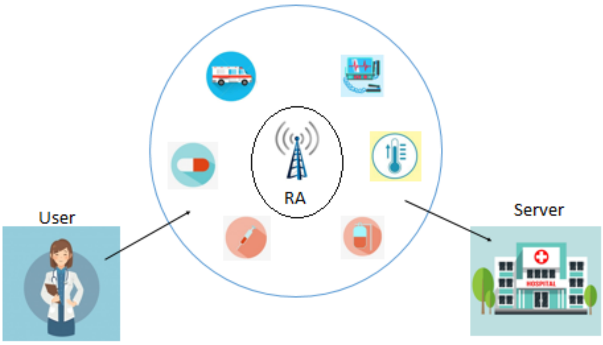

Figure 2. Overview of proposed scheme

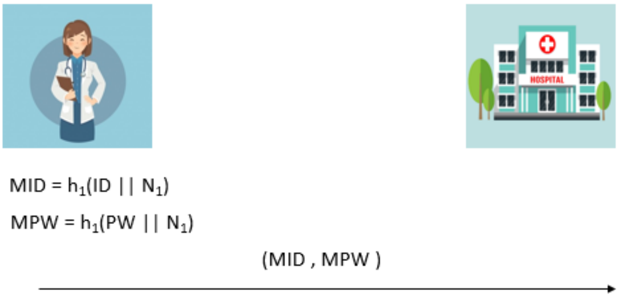

Figure 3. Registration phase of proposed scheme

4.1 Registration

First, the user selects his/her id and password to register on the network, then he/she should select a random number,N_1, and calculates $MID = h_1(ID \| N_1)$ and $MPW = h_1(pw \| N_1)$. Finally, he/her sends (MID, MPW) to the server via secure channel in order to register on the network.

4.2 Authentication

This phase is executed between user and other entities in order that authenticate each other and agree on a session key for the secure message transmission. The authentication is done via the following nine steps.

(1) User to t things: A user sends requests for access to t things with session id and user id, (s_1, MID).
(2) t Things to RA: Each thing sends a request to the RA to authenticate a user with session id, user id and it 's public point, $(s_1, MID, p_1), (s_1, MID, p_2), \ldots, (s_1, MID, p_t)$.
(3) RA to server: The RA stores the identity of the user with information related to things to which the user wants to access and forwards session

id, MID, T_1, public points of t things and $C = h_1((s_1, MID, p_1, \ldots, p_t), k, T_1)$ as request to the server, $((s_1, MID, p_1, p_2, \ldots, p_t), T_1, C)$.

(4) Server: Server performs time stamp check on received T_1 with current time stamp T_1', i.e. $|T_1' - T_1| < \Delta T$. If it is incorrect, the process is ended. Otherwise, server computes $C' = h_1((s_1, MID, p_1, \ldots, p_t), k, T_1)$ by using the k and values received from the RA. If $C' = C$ is incorrect, the server rejects the request. Otherwise, RA is authenticated for the server. Then, if MID is contained in the eligible user 's list and is allowed to access the requested things, the server accepts the request and then selects a random number $r_1 \in Z_q^*$ and calculates a polynomial $f_u(x)$ by using points $(p_1, p_2, \ldots, p_t, h_2(MPW \| r_1))$. $f_u(x)$ is a Shamir's secret sharing polynomial [28]. Our scheme uses this polynomial for authentication between the RA and the user. Then server selects a random number r_s and calculates $X_s = r_s \oplus MPW$, $MID_{new} = h_1(MID \oplus r_s)$. Finally server updates user id and stores MID_{new} in database.

(5) Server to user: The server sends session id , random number r_1 and X_s to the user,$(s_1$,r_1,$X_s)$.

(6) Server to RA: Server selects a random point p_r on the polynomial $f_u(x)$ and encrypts (s_1, p_r) using key k (the shared key between server and RA) then sends $(s_1, E_k(s_1, p_r))$ to the RA.

(7) User to RA: First, the user selects a random number r_a and computes $r_a.P$, then he/she computes the polynomial $f_u(x)$ using public points $(p_1, p_2, \ldots, p_t)$, MPW and the random number r_1. Then generates an authentication value $AUTH_1 = h_1(s_1, r_aP, MID, T_2, f_u(0))$. Finally, he/she sends time stamp T_2, session id, MID, r_aP and $AUTH_1$ to the RA, $(T_2, s_1, MID, r_aP, AUTH_1)$.

(8) RA to user: RA performs time stamp check on received T_3 with current time stamp T_2', i.e. $|T_2' - T_2| < \Delta T$? If it is incorrect, the process is ended. If it is correct, RA calculates $f_u(x)$ by using $(p_1, p_2, \ldots, p_t, p_r)$ then computes the $AUTH_1' = h_1(r_aP, T_2, f_u(0), s_1, MID)$ and compares with $AUTH_1$. If $AUTH_1' = AUTH_1$ is correct, the user is authenticated for the RA, then RA chooses a random number r_b and calculates $AUTH_2 = h(r_aP, r_bP, r_ar_bP, f_u(0))$ and sends $AUTH_2$ and r_bP to user,$(r_bP, AUTH_2)$. Either failed check leads to the rejection of the session.

(9) First, the user calculates the value of $AUTH_2' = h_1(r_aP, r_bP, r_ar_bP, f_u(0))$ and then compares it to the $AUTH_2$ sent by RA. If $AUTH_2' = AUTH_2$, RA is authenticated to the user. Finally user calculates $MID_{new} = h_1(MID \oplus r_s)$ by computing $r_s = MPW \oplus X_S$ and uses MID_{new} in the next session.

When the user and RA authenticate each other, they calculate the session key $SK = h(r_aP, r_bP, r_ar_bP, f_u(0))$ and encrypt their messages via this session key. Moreover, the details of our scheme are presented in Figure 3 and Figure 4.

5 Security and Performance Analysis

In this section, we first state our scheme security features and show that it satisfies all security requirements considered for IoT authentication scheme. Then, we present the formal security analysis of our scheme via ProVerif. Moreover, the security and performance comparisons with some previous related schemes are provided.

5.1 Informal Security Analysis

This section analyzes the security properties of the proposed scheme and shows that all security requirements stated in Table 2 are achieved in our scheme.

5.1.1 Resist to The DoS Attack

The proposed scheme is resistant to the denial-of-service attacks. In this scheme each transmitted message has a time stamp. Whenever the RA or the server receives a message, they first check the timestamps sent along with the message to prevent DoS attacks. For example, when the server receives a message, it checks $|T_1' - T_1| < \Delta T$. In this way, every timed out message is easily detected and rejected.

5.1.2 Mutual Authentication

In this scheme, the server and RA, as well as RA and user authenticate each other.The server and RA authenticate each other by the shared key k. RA calculates $AUTH_1' = h_1(r_aP, T_2, f_u(0), s_1, MID)$. If this value is equal to $AUTH_1$ sent by the user, the user is authenticated to RA. The user also calculates $AUTH_2' = h(r_aP, r_bP, r_ar_bP, f_u(0))$. If this value is equal to $AUTH_2$ sent by the RA, the RA is authenticated to the user.

5.1.3 Anonymity

In our scheme, the U_i masks the real identity ID_i with the hash of a random number N_1 in the registration phase, $MID = h_1(ID \| N_1)$ and the real identity of the user, ID_i, is never sent. So, because of using secure one way hash function h_1 and random number N_1, the user real identity can not be extracted from the MID. Also, in each session, the server gener-

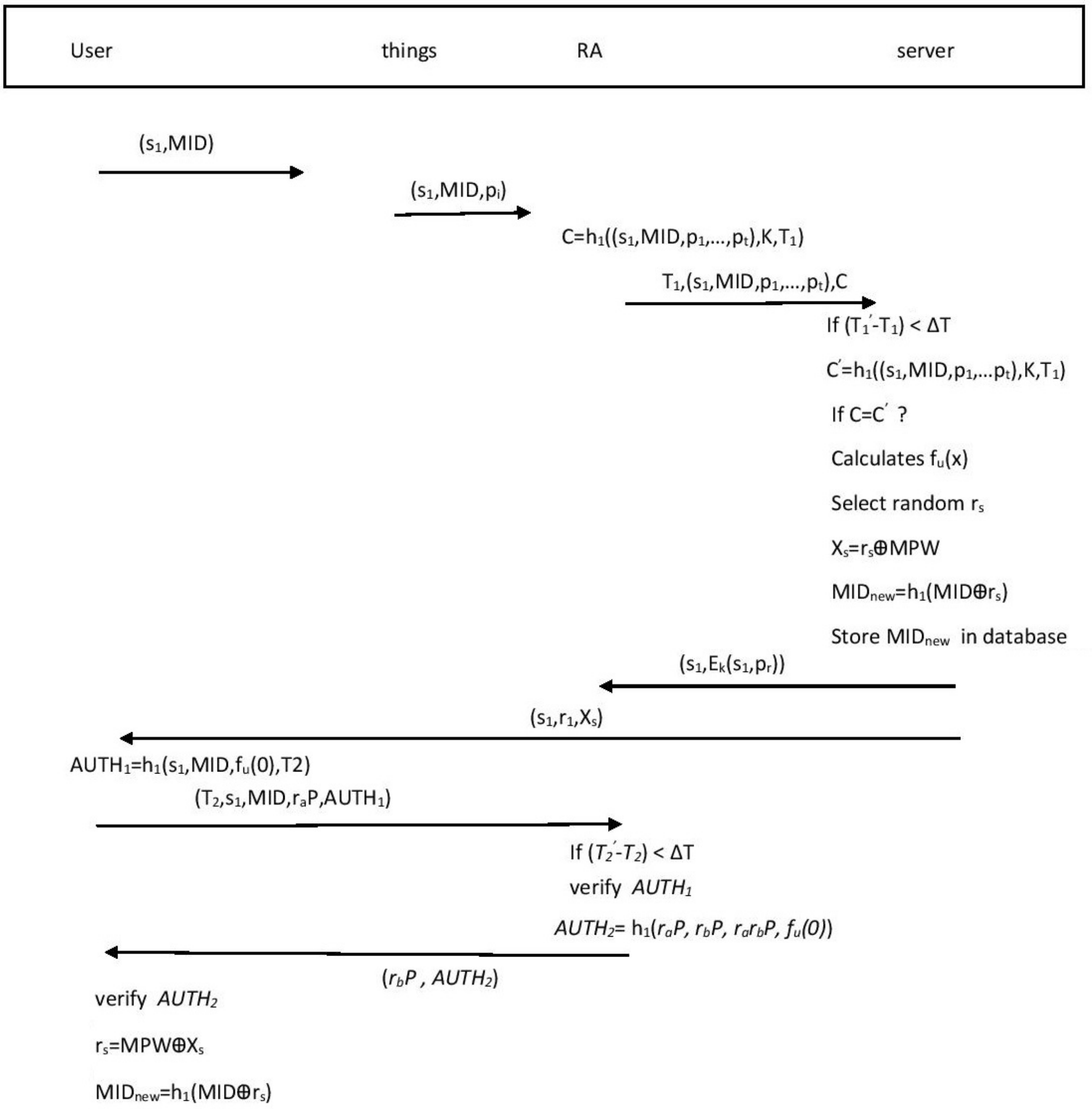

Figure 4. Authentication phase of proposed scheme

ates MID_{new} with a random number r_s and assigns $MID_{new} = MID \oplus r_s$ to the user for next session. For this reason, the user remains untraceable. Thus, our scheme provides the user anonymity and untraceability.

5.1.4 Forward Security

Forward security means that the attacker cannot find session keys created in past sessions even if he/she discovers the secret key k, obtains p_r and computes $f_u(0)$. In the proposed scheme $SK = h(r_aP, r_bP, r_ar_bP, f_u(0))$ and due to the elliptic curve Diffie-Hellman problem the attacker cannot computes r_ar_bP. So he/she cannot compute the previous session keys. Therefore, the proposed scheme can provide the forward secrecy.

5.1.5 Online/offline Password Guessing Attack

Because of the random number r_i in polynomial $f_u(x)$, this polynomial and the authentication values $AUTH_1$ and $AUTH_2$ are refreshed in each session. So the adversary cannot access any information of the user's password, even with the values of $AUTH_1$ and $AUTH_2$. Also, the $(id, pass)$ masked in the list on the server and attacker cannot access the user's passwords.

5.1.6 Non-manipulation

for computing $f_u(0)$, the MPW and random number r_i are required and only the user has them. Also, RA can calculate the polynomial $f_u(0)$ because of p_r. In this case, if the user uses an one-time password generator with the server, the adversary cannot know r_i as well, thus adversary cannot calculate the polynomial $f_u(0)$ even if they know user's password. Thus, this scheme ensures non-manipulation.

5.1.7 Man-in-the-Middle Attack

In this scheme, the adversary can obtain information such as the random number r_i, session number s_1, user id MID combination of things $(p_1, \ldots, p_t)$ and authentication values $AUTH_1$ and $AUTH_2$, but despite having access to this information, it cannot reconstruct polynomial $f_u(0)$ and manipulate the authentication values $AUTH_1$ and $AUTH_2$. So the proposed scheme can resist the man-in-the-middle attack.

5.1.8 Impersonation Attack

In this scheme, the adversary needs the user's password to calculate the polynomial $f_u(x)$ and impersonate a valid user. Due to the adversary has no access to the user's password so, it cannot impersonate valid users.

5.1.9 Verifiability, Undeniability and Unforgeability

RA can verify the authentication value $AUTH_1$ by calculating $f_u(x)$ and the adversary has no access to MPW and p_r. he user also verifies the authentication value $AUTH_2$ by calculating $r_a r_b P$ and due to the ECDHP, just user and RA can calculate $r_a r_b P$. So, verifiability is guaranteed. In addition, the users can not deny their effort for access because nobody but themselves can calculate $AUTH_1$. Furthermore, nobody can forge the user. Thus, undeniability and unforgeability are guaranteed.

Based on the above discussion, the security requirements comparison between our scheme and some related schemes listed in Table 2. As Table 2 shows, our scheme is secure against various attacks. The proposed scheme provides a greater result over some related schemes with respect to security strength.

5.2 Formal Analysis with ProVerif

Verification of security protocols is a very important and hot research area. Formal verification is a common approach to analyze security protocols. However, this method is so difficult, complex and error-prone. In order to reduce the errors, difficulty and gain more confidence on the results of the analysis, automatic verification tools have been developed with the formal method. ProVerif as a novel automatic verifier of cryptographic protocols, verifies the security properties of them such as secrecy, authentication and anonymity under the assumption of idealized primitives. Being developed by Blanchet *et al.* [29], it had been used successfully to analyze the security of cryptographic protocols of electronic voting or key exchange [30]. In ProVerif, protocols are analyzed and checked using the syntax of applied π calculus of the Blanchet *et al.* [31]. This syntax is based on the pi calculus with a rich term algebra to model cryptographic primitives [32]. It takes the model of security protocol under the syntax of applied π-calculus, in addition to the security properties we want to analyze. The security properties are analyzed in the form of queries. This tool is used in the verification of many authentication schemes [1, 33, 34]. In the following we demonstrate how to use queries for different security properties:

5.2.1 Secrecy

Checking the secrecy of messages is simply done via the following query: **query** attacker $:< message >$. This query is used to check the confidentiality of $< message >$. The query is failed if the adversary has a way to learn the value of $< message >$.

5.2.2 Authentication

Authentication generally means that if an entity A is in contact with another party B, the party (B) should be in contact with A as well. The two parties should share the same values of parameters as well. Authentication simply means that if a party A thinks that he/she is in communication with another party (B), that party (B) should also be in contact with A. This property is checked using the following queries:

- **query** ev $:< event1 > ==>$ ev $: < event2 >$. This query is important to check the authentication of a party to another one. In particular, it checks if for the occurrence of event $< event1 >$, the event $< event2 >$ has occurred at least once before. These events are executed before or after the receiving or transmission of messages. For example in the authentication between two parties A, B it may be checked that if A receives the message $< message1 >$, then B has sent the message $< message2 >$ before.
- **query** evinj $:< event1 > ==>$ evinj $:< event2 >$. This query is similar to the above query in addition to the fact that for every occurrence of $< event2 >$, there should be at exactly one occurrence of $< event1 >$.

5.2.3 Perfect Forward Secrecy

Perfect forward secrecy (PFS) is an important security requirement of cryptographic protocols. A perfect

Table 2. Security requirements comparison

Security requirements	[13]	[14]	[15]	[16]	[24]	our
Resistance to denial-of-service attack	Yes	Yes	Yes	Yes	No	Yes
Resistance to replay attack	Yes	Yes	Yes	Yes	No	Yes
Mutual authentication	No	Yes	Yes	Yes	No	Yes
Resistance to impersonation attack	Yes	Yes	Yes	Yes	Yes	Yes
Resistance to man-in-the-middle attack	Yes	Yes	Yes	Yes	Yes	Yes
Resistance to online/offline password guessing attack	Yes	Yes	Yes	Yes	Yes	Yes
Non-manipulation	Yes	Yes	Yes	Yes	Yes	Yes
Anonymity	Yes	Yes	No	Yes	No	Yes
Forward security	Yes	Yes	Yes	Yes	No	Yes
IoT based medical system	No	No	No	No	Yes	Yes

secure scheme should disclose previous session keys, when a long term secret is compromised. In order to check this property in ProVerif, each session of the scheme is represented by a phase. The compromise of the long-term key is modeled as we send out this key on the public channel in phase 1, while the main protocol is executed in phase 0. Then we check the secrecy of session keys. For example, in the following we investigate the perfect secrecy:

```
Phase 1; out (c,sk);
```

5.2.4 Verification of the Proposed Scheme with ProVerif

In this section, we verify the security of our proposed scheme with respect to key secrecy, mutual authentication of RA and server as well as mutual authentication of user and RA, anonymity and forward secrecy of session keys.

Session Key Secrecy

Session key secrecy guarantees that the attacker has no way to obtain the value of the session keys. In this scheme, the session keys of the user and RA should not be disclosed to the adversary. In this regard we check the secrecy of the session keys through the following queries:

```
query attacker (sku);
query attacker (skR);
```

Mutual Authentication of RA-User-Server

In order to verify mutual authentication of RA-User-Server, we check the correspondence of events. The mutual authentications include:

Mutual Authentication of RA-User

In order to verify the authentication of the RA to User, we execute the following query:

```
query event RAauthenticated (sku) ==>
    event RAauthentication (skR).
```

Where the event RAauthenticated is executed by the User after successfully authenticating the RA, and event RAauthentication is executed by the RA. In order to verify the authentication of the User to RA, we execute the following query:

```
query event Userauthenticated (skR) ==>
    event Userauthentication (sku).
```

Where the event Userauthenticated is executed by the RA after successfully authenticating the User, and event Userauthentication is executed by the User. The result confirms the mutual authentication of the RA to the User.

Mutual authentication of RA-Server

In order to verify the authentication of the RA to Server, we execute the following query:

```
query event RAauthenticated (skS) ==>
    event RAauthentication (skR).
```

Where the event RAauthenticated is executed by the Server after successfully authenticating the RA, and event RAauthentication is executed by the RA. In order to verify the authentication of the server to RA, we execute the following query:

```
query event Serverauthenticated (skR) ==>
    event Serverauthentication (skS).
```

Where the event Serverauthenticated is executed by the RA after successfully authenticating the server, and event Serverauthentication is executed by the RA. The result confirms the mutual authentication of the RA to the server.

User Anonymity

Anonymity of an entity implies that the identity of that entity should be hidden from adversary. In order to verify the anonymity of the user with identity IDi, the following query is executed:

```
query attacker (IDi);
```

Perfect Forward Secrecy

In order to ensure the Perfect Forward Secrecy of the proposed scheme, the secrecy of the session keys (sks,skR,sku) should be guaranteed in previous sessions, if the secret key k gets revealed to the adversary. In this regard, this secret key k is given to the attacker in phase 1 , whereas the secrecy of the session keys are checked in phase 0:

```
Phase 1 ; out(c, k);
query attacker (sks);
query attacker (skg);
query attacker (sku);
```

5.3 Performance Analysis

In this section,we compare the performance of our scheme with some previously related schemes to manifest the merits of the proposed scheme.

In order to carry out the performance comparison, we define some notations as follows:

- T_H is the execution time of a hash operation.
- T_E is the execution time of a symmetric encryption/decryption.
- T_S is the execution time of an ECC point multiplication operation.

The experiment results in [35] is utilized here. In [35] authors executed the experiment on Ubuntu operating system via the PBC library on a system with RAM size 2048 MB and 2.20 GHZ Dual CPU E2200. Computation time of each cryptographic operation, given the above system specifications, is as follow: $T_H \approx 0.0023ms$,$T_E \approx 0.0046ms$ and $T_S \approx 2.226ms$. The computational costs of each scheme are presented in Table 3. In this table, for each of the schemes, the computational cost is calculated in case of user, RA and server. As indicated in Table 3 and Figure 5, our proposed scheme has less computational cost in comparison with [13–16] and [24]. Table 3 demonstrates that the computational cost of the user and RA entities in our scheme is slightly larger than [24] which is reasonable because of, as shown in Table 2, our scheme more secure than [24] scheme.

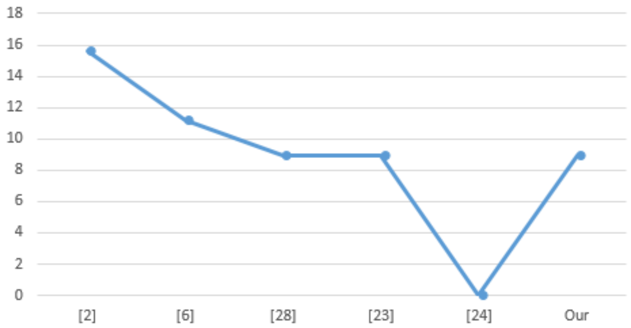

Figure 5. Computational cost comparison

6 Conclusions

IoT development during recent years has significantly influenced the health industry and caused to present several more efficient medical services in it. Due to the sensitivity of medical information, the users authentication is one of the major challenge in IoT based medical system. In this paper, we presented an improved scheme for mutual authentication in health care systems. The proposed scheme provides mutual authentication between the user and RA, protects the users anonymity and is resistant to various security attacks. In addition, using ProVerif, it was formally proved that our scheme is resistant to numerous passive and active attacks. Security and efficiency comparisons indicate that our scheme is more applicable and secure than previous related ones.

References

[1] Aida Akbarzadeh, Majid Bayat, Behnam Zahednejad, Ali Payandeh, and Mohammad Reza Aref. A lightweight hierarchical authentication scheme for internet of things. *Journal of Ambient Intelligence and Humanized Computing*, pages 1–13, 2018.

[2] Daewon Lee and HwaMin Lee. Iot service classification and clustering for integration of iot service platforms. *The Journal of Supercomputing*, 74(12):6859–6875, Dec 2018.

[3] Isabel de la Torre Díez, Susel Góngora Alonso, Sofiane Hamrioui, Eduardo Motta Cruz, Lola Morón Nozaleda, and Manuel A. Franco. Iot-based services and applications for mental health in the literature. *Journal of Medical Systems*, 43(1):11, Dec 2018.

[4] Wei-Liang Tai, Ya-Fen Chang, and Ya-Ling Lo.

Entities	[13]	[14]	[15]	[16]	[24]	our
user	$7T_H + 4T_S + 2T_E$ (8.9293 ms)	$7T_H + 4T_S + 2T_E$ (8.9293 ms)	$2T_S + 8T_H$ (4.4704)	$2T_S + 11T_H$ (4.4773)	$2T_H$ (0.0046 ms)	$4T_H + 2T_S$ (4.4612 ms)
RA	- -	- -	- -	- -	T_E (0.0046 ms)	$2T_H + 2T_S + T_E$ (4.4612 ms)
Server	$7T_H + 3T_S$ (6.6987 ms)	$T_S + 4T_H + T_E$ (2.2398 ms)	$2T_S + 5T_H$ (4.4635)	$2T_S + 2T_E + 8T_H$ (4.4796)	$T_H + T_E$ (0.0069 ms)	$2T_H + T_E$ (0.0092 ms)

Table 3. Computational cost comparison

An anonymity, availability and security-ensured authentication model of the iot control system for reliable and anonymous ehealth services. *Journal of Medical and Biological Engineering*, Jan 2018.

[5] V. Jagadeeswari, V. Subramaniyaswamy, R. Logesh, and V. Vijayakumar. A study on medical internet of things and big data in personalized healthcare system. *Health Information Science and Systems*, 6(1):14, Sep 2018.

[6] B. Lakshmi Dhevi, K. S. Vishvaksenan, K. Senthamil Selvan, and A. Rajalakshmi. Patient monitoring system using cognitive internet of things. *Journal of Medical Systems*, 42(11):229, Oct 2018.

[7] Bahar Farahani, Farshad Firouzi, Victor Chang, Mustafa Badaroglu, Nicholas Constant, and Kunal Mankodiya. Towards fog-driven iot ehealth: Promises and challenges of iot in medicine and healthcare. *Future Generation Computer Systems*, 78:659–676, 2018.

[8] P. Mohamed Shakeel, S. Baskar, V. R. Sarma Dhulipala, Sukumar Mishra, and Mustafa Musa Jaber. Maintaining security and privacy in health care system using learning based deep-q-networks. *Journal of Medical Systems*, 42(10):186, Aug 2018.

[9] Qi Jiang, Jianfeng Ma, Zhuo Ma, and Guangsong Li. A privacy enhanced authentication scheme for telecare medical information systems. *Journal of Medical Systems*, 37(1):9897, Jan 2013.

[10] Saru Kumari, Muhammad Khurram Khan, and Rahul Kumar. Cryptanalysis and improvement of ‘a privacy enhanced scheme for telecare medical information systems’. *Journal of medical systems*, 37(4):9952, 2013.

[11] Hung-Ming Chen, Jung-Wen Lo, and Chang-Kuo Yeh. An efficient and secure dynamic id-based authentication scheme for telecare medical information systems. *Journal of medical systems*, 36(6):3907–3915, 2012.

[12] Tianjie Cao and Jingxuan Zhai. Improved dynamic id-based authentication scheme for telecare medical information systems. *Journal of Medical Systems*, 37(2):9912, Jan 2013.

[13] Ruhul Amin, Sk Hafizul Islam, GP Biswas, Muhammad Khurram Khan, and Neeraj Kumar. An efficient and practical smart card based anonymity preserving user authentication scheme for tmis using elliptic curve cryptography. *Journal of medical systems*, 39(11):180, 2015.

[14] Shehzad Ashraf Chaudhry, Muhammad Tawab Khan, Muhammad Khurram Khan, and Taeshik Shon. A multiserver biometric authentication scheme for tmis using elliptic curve cryptography. *Journal of medical systems*, 40(11):230, 2016.

[15] Shuming Qiu, Guoai Xu, Haseeb Ahmad, and Licheng Wang. A robust mutual authentication scheme based on elliptic curve cryptography for telecare medical information systems. *IEEE access*, 6:7452–7463, 2017.

[16] Arezou Ostad-Sharif, Dariush Abbasinezhad-Mood, and Morteza Nikooghadam. A robust and efficient ecc-based mutual authentication and session key generation scheme for healthcare applications. *Journal of medical systems*, 43(1):10, 2019.

[17] Siwei Peng. An id-based multiple authentication scheme against attacks in wireless sensor networks. In *Cloud Computing and Intelligent Systems (CCIS), 2012 IEEE 2nd International Conference on*, volume 3, pages 1042–1045. IEEE, 2012.

[18] Wenbo Shi and Peng Gong. A new user authentication protocol for wireless sensor networks using elliptic curves cryptography. *International Journal of Distributed Sensor Networks*, 9(4):730831, 2013.

[19] Hung-Min Sun, Bing-Zhe He, Chien-Ming Chen, Tsu-Yang Wu, Chia-Hsien Lin, and Huaxiong Wang. A provable authenticated group key agreement protocol for mobile environment. *Information Sciences*, 321:224–237, 2015.

[20] Parikshit N Mahalle, Neeli Rashmi Prasad, and Ramjee Prasad. Novel threshold cryptography-

based group authentication (tcga) scheme for the internet of things (iot). 2014.

[21] Pawani Porambage, Corinna Schmitt, Pardeep Kumar, Andrei Gurtov, and Mika Ylianttila. Pauthkey: A pervasive authentication protocol and key establishment scheme for wireless sensor networks in distributed iot applications. *International Journal of Distributed Sensor Networks*, 10(7):357430, 2014.

[22] Boyi Xu, Li Da Xu, Hongming Cai, Cheng Xie, Jingyuan Hu, Fenglin Bu, et al. Ubiquitous data accessing method in iot-based information system for emergency medical services. *IEEE Trans. Industrial Informatics*, 10(2):1578–1586, 2014.

[23] Jia-Li Hou and Kuo-Hui Yeh. Novel authentication schemes for iot based healthcare systems. *International Journal of Distributed Sensor Networks*, 11(11):183659, 2015.

[24] YoHan Park and YoungHo Park. A selective group authentication scheme for iot-based medical information system. *Journal of medical systems*, 41(4):48, 2017.

[25] Yanxiao Liu, Qindong Sun, Yichuan Wang, Lei Zhu, and Wenjiang Ji. Efficient group authentication in rfid using secret sharing scheme. *Cluster Computing*, pages 1–7, 2018.

[26] Neal Koblitz. Elliptic curve cryptosystems. *Mathematics of computation*, 48(177):203–209, 1987.

[27] Steven D Galbraith. *Mathematics of public key cryptography*. Cambridge University Press, 2012.

[28] Adi Shamir. How to share a secret. *Communications of the ACM*, 22(11):612–613, 1979.

[29] Bruno Blanchet, Ben Smyth, and Vincent Cheval. Proverif 1.93: Automatic cryptographic protocol verifier, user manual and tutorial. *Internet][cited June 2016], Available from: https://www. bensmyth. com/publications/2010-ProVerif-manualversion-1.93*, 2016.

[30] Maria Christofi and Aline Gouget. Formal verification of the mera-based eservices with trusted third party protocol. In *IFIP International Information Security Conference*, pages 299–314. Springer, 2012.

[31] Stephanie Delaune, Mark Ryan, and Ben Smyth. Automatic verification of privacy properties in the applied pi calculus. pages 263–278, 2008.

[32] Bruno Blanchet. Automatic verification of security protocols in the symbolic model: The verifier proverif. In *Foundations of Security Analysis and Design VII*, pages 54–87. Springer, 2014.

[33] Seyed Morteza Pournaghi, Behnam Zahednejad, Majid Bayat, and Yaghoub Farjami. Necppa: A novel and efficient conditional privacy-preserving authentication scheme for vanet. *Computer Networks*, 134:78–92, 2018.

[34] Behnam Zahednejad, Mahdi Azizi, and Morteza Pournaghi. A novel and efficient privacy preserving tetra authentication protocol. In *2017 14th International ISC (Iranian Society of Cryptology) Conference on Information Security and Cryptology (ISCISC)*, pages 125–132. IEEE, 2017.

[35] H Hakan Kilinc and Tugrul Yanik. A survey of sip authentication and key agreement schemes. *IEEE Communications Surveys & Tutorials*, 16(2):1005–1023, 2014.

Mahdieh Ebrahimi received M.Sc. degree in secure computing from the department of computer engineering, Shahed University, Tehran, Iran. Her research interests include Cryptography and IoT security.

Majid Bayat is an assistant professor of computer engineering of Shahed University, Tehran, Iran. His research interests include IoT security. He has authored over 40 papers in international journals and conferences in the above areas.

Behnam Zahednejad is currently a Ph.D. student of Guangzhou University, Guangzhou, China. His research interests are in the areas of Cryptography, Security protocols, Formal analysis and Information security.