SYSTEMS-LEVEL QUALITY IMPROVEMENT

A Standard Mutual Authentication Protocol for Cloud Computing Based Health Care System

Prerna Mohit[1] · Ruhul Amin[2] · Arijit Karati[1] · G. P. Biswas[1] · Muhammad Khurram Khan[3]

Received: 24 August 2016 / Accepted: 5 February 2017 / Published online: 17 February 2017

Abstract Telecare Medical Information System (TMIS) supports a standard platform to the patient for getting necessary medical treatment from the doctor(s) via Internet communication. Security protection is important for medical records (data) of the patients because of very sensitive information. Besides, patient anonymity is another most important property, which must be protected. Most recently, Chiou et al. suggested an authentication protocol for TMIS by utilizing the concept of cloud environment. They claimed that their protocol is patient anonymous and well security protected. We reviewed their protocol and found that it is completely insecure against patient anonymity. Further, the same protocol is not protected against mobile device stolen attack. In order to improve security level and complexity, we design a light weight authentication protocol for the same environment. Our security analysis ensures resilience of all possible security attacks. The performance of our protocol is relatively standard in comparison with the related previous research.

Keywords Anonymity · Signature · Cloud database · TMIS

This article is part of the Topical Collection on *Systems-Level Quality Improvement*

✉ Ruhul Amin
amin_ruhul@live.com

Prerna Mohit
prernamohit@outlook.com

Arijit Karati
arijit.karati@gmail.com

G. P. Biswas
gpbiswas@gmail.com

Muhammad Khurram Khan
mkhurram@ksu.edu.sa

[1] Department of Computer Science and Engineering, Indian Institute of Technology (ISM), Dhanbad, 826004, Jharkhand, India

[2] Department of Computer Science and Engineering, Thapar University, Patiala, 147004, Punjab, India

[3] Center of Excellence in Information Assurance, King Saud University, Riyadh, Saudi Arabia

Introduction

In medical-system environment, the cloud users store medical records in the cloud database to retrieve the medical information safely. As it is known that cloud server is not fully trusted, hence a secure and authentication protocol is needed to resist common security attacks [28]. In recent years there are some authentication protocols [1–3, 6, 12, 38] proposed for TMIS, where the patients get their treatment online. As mentioned in [7] Telecare medical information system (TMIS) facilitates medical practitioners and patients to establish a communication over public network to provide health care services directly in the patient's home. As an explanation of TMIS, it is observed that both patients and doctors can work together via TMIS server, i.e. a patient upload diagnosis symptoms to the server and a doctor collects them and submits diagnosis report to the server as if they are interacting directly and this is happened through TMIS. Moreover, how to get medical resources more conveniently and securely are the major concerning issues as the communication is performed over the public channel. In addition, the security requirements, data confidentiality, data/patient authentication and patient anonymity are the important features to be maintained during the communication. In order to maintain patient anonymity [13, 19, 35], the true identity of the patient need to be hidden from the others including eavesdropper. The patients' diagnosis reports in TMIS are very important and they should not be disclosed publicly.

As the information shared between cloud server and patients/doctors are very critical information and thus, data should be stored securely. As medical data come under crucial data and failure of it may cause failure of ones life [44], so it is necessary to develop a secure protocol so that no adversary can try to obtain patients' medical records and misuse it. Recently, there have been some protocols presented to realize anonymity issue. Most of these existing protocols may not be applicable to provide patient anonymity in healthcare environment.

Motivation and contribution

Recently, Chiou et al. [10] devised an authentication protocol for TMIS that can be used in the cloud environment. It is reviewed and shown that

- Their protocol doest no support patient anonymity.
- The protocol does not provide security against mobile device stolen attack.

In order to achieve security against the aforementioned attacks and to provide a complete package, a light-weighted authentication protocol for TMIS is proposed which is suited for the cloud environment. Our protocol has several important features, which are discussed below:

- *Mutual authentication* is achieved between healthcare center and cloud server, patients and cloud server, and doctor and cloud server to strength the security of a system and transmitting information.
- *Patient anonymity* is supported during the data transmission by securing patient identity.
- The protocol resists strong security attacks, i.e., provides security against patient *anonymity, non-repudiation* and *confidentiality* of data.
- The protocol is also free from some known security attacks which are discussed in "Security analysis".
- We compare our protocol with other existing protocols and found that it achieves minimum *computational* and *communicational overheads*.

Related works

Smart card based authentication mechanism is the most common technique which is used to prevent unauthorized access over the insecure networks. There are so many authentication protocols [30, 31, 36] available using smart card [24], where the user chooses a password and carries a smart card with it. The authentication protocol is very useful in various applications, such as ad-hoc network , wireless sensor network , and medical system [4, 5, 14–18, 20, 29, 34]. Wu et al. first presented a password-based user authentication scheme for medical system and web-based hospital-acquired infection surveillance information system (WHISS) [40, 42]. After that, Wu-Lee et al. [41] proposed a new scheme using smartcard and password-based security protocol for TMIS. Then, He et al. [11] pointed out that Wu et al.'s scheme [41] has various security loopholes, such as impersonation attack and insider attack. They also suggested an enhanced protocol. In 2012, Wei et al. [39] recognized that both the existing protocols [11, 41] which are not protected against security attacks and suggested an enhanced scheme to resist the existing attacks. After that, Zhu et al. [45] presented that Wei et al.'s protocol is not secured against off-line password guessing attack and provided an extended authentication scheme suitable for TMIS, which is based on the RSA cryptosystem. However, there are several security weaknesses that have been identified by Khan et al. [23] on the scheme [45]. After that, Khan et al. designed an improved version of Zhu et al.'s [45] scheme. In 2013, Jiang et al. [21] discussed a strong chaotic map-based authentication scheme with anonymity for Telecare Medical Information System. Kumari et al. [25] claimed that their scheme fails to provide impersonation attack, online password guessing attack and stolen-verifier attack. According to the above reasons, the proposed scheme should defend against these weaknesses. However, Mishra et al. [33] discussed that the protocol in [21] does not provides denial-of-service attack.

In 2013, Tan et al. [37] proposed a smartcard based password authentication and key agreement scheme by applying biometric technique and pointed out that the scheme is more standard and secured. However, Yan et al. [43] demonstrated that the protocol [37] fails to resist Denial-of-Service attack. In order to resolve the problem, Yan et al. [43] proposed a improved authentication scheme to overcome the drawbacks of [37]. In 2014, Mishra et al. [32] described that Yan et al. [43] protocol have a number of security loopholes, such as the user privacy problem, inefficient login phase, inefficient password and biometric update phase, password guessing attack and three-factor authentication problem. To resolve the above mentioned problems, they also proposed an improved protocol. Li et al. [27] claimed that the Lee et al.'s [26] chaotic-maps based user authentication scheme has security flaws such as lack of user identity in the authentication phase, service misuse attacks and suggested a more efficient solution for obtaining the medical system.

In 2014 Chen et al. [9] combines the cloud computing with mobile devices to provide medical resources and uses cryptographic technology to protect the patients personal information. However, the protocol has several weaknesses. Chen et al. [8] also discussed another protocol for medical system based on cloud computing, but the protocol does not support patient anonymity and message authentication. In order to solve the problem of [8], Chiou et al. [10] modified the existing scheme and claimed that the protocol provides real tele-medicine service, patient anonymity, and message authentication.

Road map of the paper

"Introduction" gives the introduction of TMIS, followed by study of Chiou et al.'s protocol in "Review of Chiou et al.'s scheme [10]". After that, cryptanalysis of Chiou et al.'s is presented in "Cryptanalysis of Chiou et al.'s scheme". "Proposed protocol" discusses the proposed protocol for medical system over cloud server, followed by its security analysis in "Security analysis". Performance evaluation is given in "Performance analysis". Finally, the conclusion of this paper is given in "Conclusion".

Review of Chiou et al.'s scheme [10]

Chiou et al. [10] proposed an improved version of cloud-based privacy, authentication scheme for medical treatment. There are five entities, e.g. Patient, Doctor, Cloud, Healthcare Center and Body Sensor. The Chiou's scheme consists of four different phases, e.g. (1) Healthcare center upload phase, (2) Patient data upload phase, (3) Treatment phase and (4) Checkup phase. All the notations are represented in Table 1.

Table 1 Symbols used

Symbol	Description
$E_x(m)$	Encryption of message m using key x
$D_x(m)$	Decryption of message m using key x
ID_i	Unique identity of i
PU_x	Public key of entity x
PR_x	Private key of entity x
SK_{xy}	Session key between x and y
$h(.)$	One-way cryptographic hash function.
$X\|\|Y$	X concatenate with Y
$X =?Y$	Whether X equal Y or not
$S_k(m)$	Use key k to sign m
$V_k(m)$	Use key k to verity m
key_P	The pre-generated key of patient
MD_x	Message digest of x
OTP	One-time password, generated by healthcare centre
NID	Pseudo-random number
G_{key}	The group of key

Healthcare center upload phase (HUP)

In this phase, the patient goes to the healthcare center for the treatment and the healthcare generates medical report as $m_H = (ID_P, Data_H, T_H^1)$. In addition, a pseudo-random identity (NID_P) is allotted to patients and performs mutual authentication to send the medical report to the cloud server. The operations used in this phase are as follows.

Step 1. The healthcare uses the private key to sign the report $Sig_H = S_{SK_H}(m_H)$; and encrypts m_H as $C_1 = E_{key_1}(m_H)$, where $key_1 = h(e(PK_P, SK_H), NID_P)$. Then, the healthcare randomly chooses a session key $key_{HC} \in G_{key}$ to compute $s_1 = h(e(PK_C, SK_H), T_H^1) \oplus key_{HC}$; $s_2 = h(key_{HC})$. Further, healthcare computes $C_2 = E_{key_{HC}}(ID_P, NID_P, C_1, Sig_H)$ and sends $< ID_H, s_1, s_2, C_2, T_H^1 >$ to the cloud.

Step 2. On receiving, cloud server verifies $T_C^1 - T_H^1 < \triangle T$ and computes $key'_{HC} = h(e(PK_H, SK_C), T_H^1) \oplus s_1$. Further, the server verifies the equation $s_2 ?= h(key'_{HC})$ and decrypts C_2 using key'_{HC}. Finally, cloud server stores ID_P, NID_P, C_1, Sig_H corresponding to patient and sends $s_3 = h(key_{HC} + 1)$ to healthcare.

Step 3. After receiving the message s_3 from cloud. The healthcare verifies $s_3 ?= h(key_{HC}+1)$. If it holds, the healthcare uploads the data otherwise, rejects the session.

Patient data upload phase (PUP)

Body sensor measures patients health information $m_B = (ID_P, Data_B, T_P^1)$ and gets an appointment sequence number sn. Then, it updates and uploads encrypted m_B and m_H to cloud server. The details of this phase are described below:

Step 1. The patient randomly chooses $key_{PC} \in G_{key}$ as a session key between patient and cloud. Then, patient computes $s_4 = h(e(PK_C, SK_P), T_P^1) \oplus key_{PC}$; $s_5 = h(key_{PC})$ and sends $< NID_P, s_4, s_5, T_P >$ to the cloud.

Step 2. On receiving these messages, cloud server verifies $T_C^2 - T_P^1 < \triangle T$ and computes $key'_{PC} = h(e(PK_P, SK_C), T_P^1) \oplus s_4$. Further, the cloud verifies whether the equation $s_5 ?= h(key'_{PC})$ is correct or not. If it is correct, obtains the stored data C_1 and signature Sig_H of patient using NID_P. Finally, it sends $s_6 = h(key'_{PC}, C_1, Sig_H), C_1$ and Sig_H to patient.

Step 3. On receiving, patient verifies $s_6 ?= h(key_{PC}, C_P, Sig_H)$ and computes $key_1 = h(e(PK_H, SK_P), NID_P)$ to decrypt $m_H ?= D_{key_1}(C_1)$.

After that, patient verifies signature $m_H ? = V_{PK_H}(Sig_H)$ and computes $key_2 = h(e(PK_D, SK_P), sn)$; $C_3 = E_{key_2}(m_H, m_B)$ and $s_7 = h(key_{PC}, ID_D, sn, C_3)$. Finally, the patient sends $< ID_D, sn, s_7, C_3 >$ to cloud, and renews $NID_P^{new} = h(NID_P||key_{PC})$ with updated NID_P^{new}.

Step 4. On receiving these messages, cloud verifies whether $s_7 ? = h(key'_{PC}, ID_D, sn, C_3)$ holds or not. If it does, cloud computes $NID_P^{new} = h(NID_P ||key_{PC})$ and stores $< NID_P^{new}, ID_D, sn >$, replaces C_1 with C_3; otherwise terminates the phase.

Treatment phase (TP)

Doctor obtains patient's id ID_P and sequence number sn form cloud server after establishing mutual authentication. After diagnosing Patient's symptom, doctor uploads diagnostic records to cloud. The detail is described below.

Step 1. Doctor chooses a random number $key_{DC} \in G_{key}$ as a session key between doctor and cloud. Then, Doctor computes messages $s_8 = h(e(PK_C, SK_D), T_D^1) \oplus key_{DC}$, $s_9 = h(key_{DC})$ and sends $< ID_D, s_8, s_9, T_D^1 >$ to the cloud.

Step 2. On receiving these messages, cloud verifies $T_C^3 - T_D^1 < \triangle T$ and computes $key'_{DC} = h(e(PK_D, SK_C), T_D^1 \oplus s_8)$. Further, the cloud server verifies the equation $s_9 =?h(key'_{DC})$ and computes $s_{10} = h(key'_{DC}, sn, C_3, Sig_H)$. Finally, sends $< s_{10}, sn, C_3, Sig_H >$ to the doctor.

Step 3. Upon receiving these messages, doctor verifies $s_{10} ? = h(key_{DC}, sn, C_3, Sig_H)$ and computes $key'_2 = h(e(PK_P, SK_D), sn)$ to decrypt $(m_H, m_B) = D_{key'_2}(C_3)$. Further, doctor verifies the signature by checking $m_H ? = V_{PK_H}(Sig_H)$ and makes a medical diagnosis based on the medical reports and generates medical record $m_D = (ID_P, Data_D, T_D^2)$. Finally, the doctor encrypts the report using key'_2 as $C_4 = E_{key'_2}(m_H, m_B, m_D)$ and sends $s_{11} = h(key_{DC}, C_4, Sig_D), sn, C_4, Sig_D$ to cloud.

Step 4. On receiving these messages, cloud verifies whether the equation $s_{11} =?h(key'_{DC}, C_4, Sig_D)$ holds or not. If it does, cloud stores C_4 and Sig_D otherwise, terminates the phase.

Check up phase (CP)

The patient can use his mobile phone to download the medical report generated by doctor and arrange the appropriate medical treatment based on the report.

Step 1. The patient randomly chooses a number $key_{PC} \in G_{key}$ as the session key between patient and cloud. Then, patient computes the messages $s_{12} = h(e(PK_C, SK_P), T_P^2) \oplus key_{PC}$; $s_{13} = h(key_{PC})$ and sends $< NID_P^{new}, s_{12}, s_{13}, T_P^2 >$ to cloud.

Step 2. On receiving these messages, cloud verifies $T_C^4 - T_P^2 < \triangle T$ and computes $key'_{PC} = h(e(PK_C, SK_P), T_P^2) \oplus s_{12}$. Further, the cloud verifies equation by checking $s_{13} =?h(key'_{PC})$ holds or not. If it does, cloud sends $< s_{14} = h(key'_{PC}, C_4, Sig_D), C_4, Sig_D >$ to patient.

Step 3. On receiving these messages, patient verifies $s_{14} ? = h(key_{PC}, C_4, Sig_D)$ and obtains $(m_H, m_B, m_D) = D_{key_2}(C_4)$ and also verifies the validation of Sig_D by checking $m_D ? = V_{PK_D}(Sig_D)$ and takes medical measures according to the diagnosis report, using the pre-generated key key_P to encrypt m_H, m_B, m_D, and obtains $C_5 = E_{key_P}(m_H, m_B, m_D)$. Finally, the patient sends $S_{15} = h(key_{PC}, C_5), C_5$ to cloud.

Step 4. On receiving these messages, cloud verifies whether $S_{15} =?h(key'_{PC}, C_5)$ holds or not. If it does, cloud replaces C_4 with C_5 and stores C_5 in the cloud. Depending on the need of patients, the medical staff can get the pre-generated key key_P from KGC to decrypt C_5.

Cryptanalysis of Chiou et al.'s scheme

Chen et al. [10] point out that their scheme supports user anonymity which is the most important feature for the medical system. However, we demonstrated that their scheme fails to support it. We also found that, Chiou et al.'s protocol fails to support mobile device stolen problem. The detail description is provided below.

Fails to support patients' anonymity

User anonymity states that adversary $\mathcal{A}$ should not be able to obtain the identity of user by any means of communication. Chiou et al. claim that identity of patient ID_P is protected from the adversary $\mathcal{A}$. However, we have noted that it is not true. During patient data upload phase, cloud sends $Sig_H = S_{SK_H}(m_H)$ to patient via public channel. As, the communication media is public, the adversary $\mathcal{A}$ can easily access it. For instance, assume an attacker interrupts in between patient and cloud and obtains the value of Sig_H, the attacker can apply the public key of healthcare to obtain the message as $m_H = PK_H(Sig_H) = V_{PK_H}(S_{SK_H}(m_H))$; where, m_H contains $(ID_P, Data_H, T_H^1)$. Hence, patient anonymity breaks. Similarity in the treatment phase, cloud sends $Sig_H = S_{SK_H}(m_H)$ to the doctor via open channel, where an attacker can interrupt and get the original message. Hence, the scheme fails to support patient anonymity as well as message protection.

Fails to support mobile device stolen

As mobile device of patient is an important tool for communication with doctor. The body sensor embedded in patients body provides the updated report of patient to the mobile device. Now, assume the mobile device of the patient is stolen, still the body sensor will send the message $m_B = (ID_P, Data_B, T_P^1)$ to the patient's mobile phone. As, sensor is sending the message m_B directly without any security to the mobile device of patient. Thus, the adversary $\mathcal{A}$ have direct access to the mobile phone of patient and message m_B.

Similarly, there is no security warning is given by patient to differentiate between the original patient and attacker. Hence, the scheme does not support stolen Mobile device attack.

Proposed protocol

Architecture and discussion

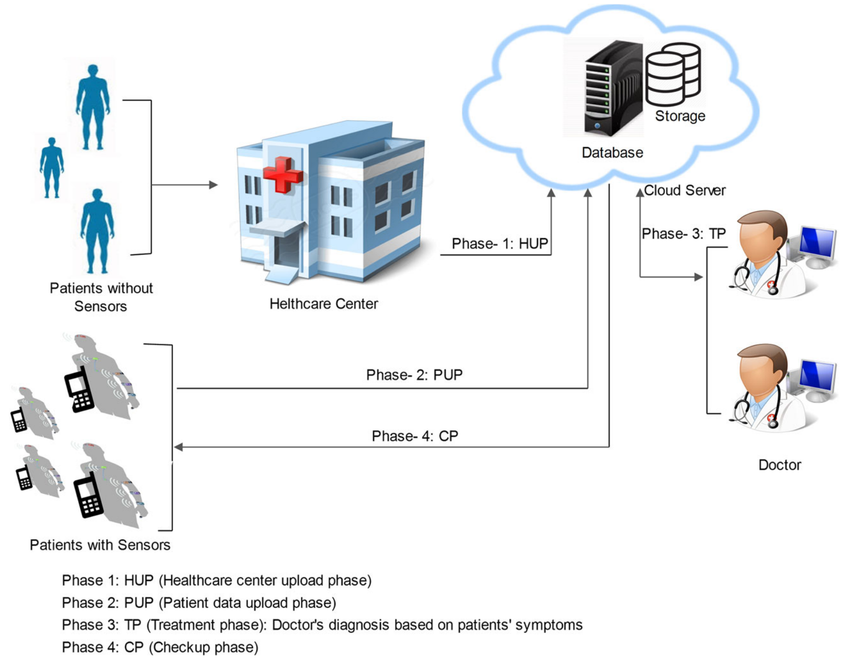
There are five entities involve in the communication: 1) Patient: A person, who is requesting for medical treatment, 2) Doctor: A person, who has been trained in medical science and whose job is to treat patients, 3) Cloud: A server to store patient's medical records, 4) Healthcare Center: Physical place where the patient receives medical treatment, 5) Body Sensor: A device connected with a physical sense of the patient. The architecture is shown in Fig 1 and explanation is given below:

- Initially the patient goes to the healthcare center for the health inspection/ routine-checkup and performs registration, where hospital maintains the report of patients.
- The health care uploads the report of a patient to the cloud.
- Body sensors embedded in the patient's body collect the health report of the patient and send to patient's mobile phone (securely).
- After that, patient uploads new report by integrating the previous report of health care with the generated report by body sensor to the cloud.
- The cloud sends the report of the patient to the respected doctor in order of sequence number.
- The doctor performs treatment by looking into the report and uploads the new report with the digital signature to the cloud server.
- The cloud sends the final report to the patient which contains the treatment of the patient.

Protocol description

This section proposes a new lighter weighted protocol for medical system which involves cloud server. Our protocol consists of four phases: 1) Healthcare center upload phase, 2) Patient data upload phase, 3) Treatment phase and 4) Check up phase.

Fig. 1 Protocol architecture and authentication process with ordering of phases


Storage
Database
Cloud Server
Patients without Sensors
Helthcare Center
Phase- 1: HUP
Phase- 3: TP
Phase- 2: PUP
Phase- 4: CP
Doctor
Patients with Sensors
Phase 1: HUP (Healthcare center upload phase)
Phase 2: PUP (Patient data upload phase)
Phase 3: TP (Treatment phase): Doctor's diagnosis based on patients' symptoms
Phase 4: CP (Checkup phase)

Healthcare center upload phase (HUP)

The Patient registers himself in healthcare center, and the healthcare center allots one-time password (OTP) and a dynamic pseudo-random identity NID to patient via mobile device. In this phase, the healthcare center performs mutual authentication with cloud and uploads patient's inspection report to cloud as shown in Fig. 2 and described below.

Step 1. The healthcare center generates inspection report $m_H = (ID_P, Data_P)$, and inputs unique identity of healthcare ID_H (either random number or MAC address of system) with a randomly selected number R. Further, the healthcare sends $< ID_H, R >$ to cloud server via secure channel.

Step 2. On receiving these messages, the cloud server computes $A = h(ID_H \parallel R \parallel x)$, $S_1 = h(A)$ and $B = ID_H \oplus x$, where x is a secret key of cloud and sends B, S_1 to healthcare via public channel.

Step 3. On receiving these messages, healthcare computes $x' = B \oplus ID_H$, $A' = h(ID_H \parallel x' \parallel R)$ and verifies whether $S_1' \ =?h(A')$ holds or not. If it does, the healthcare authenticates the cloud server and computes the session key between healthcare center and cloud as $SK_{HC} = h(ID_H \parallel A' \parallel B)$. After performing mutual authentication, the healthcare encrypts the report as $C_H = E_{key_1}(m_H)$ using $key_1 = h(ID_P \parallel OTP)$ and signs the message $Sig_H = S_{PR_H}(MD_H)$, where $MD_H = h(m_H)$ is a message digest. Further, the healthcare encrypts ID_P, C_H, Sig_H, NID with the session key SK_{HC} to get $C_1 = E_{SK_{HC}}(ID_P, C_H, Sig_H, NID)$ and $S_2 = h(SK_{HC}||C_1)$ and finally sends S_2, C_1 to cloud server via insecure channel.

Step 4. Upon receiving these messages, cloud computes $SK'_{HC} = h(ID_H \parallel A \parallel B)$ and verifies whether equation $S_2'? = h(SK'_{HC}||C_1)$ holds or not. If it does, cloud authenticates the healthcare center and decrypts the messages using the session key SK'_{HC} to obtain $(ID_P, C_H, Sig_H, NID) = D_{SK_{HC}}(C_1)$ otherwise, it fails and goes to Step 1.

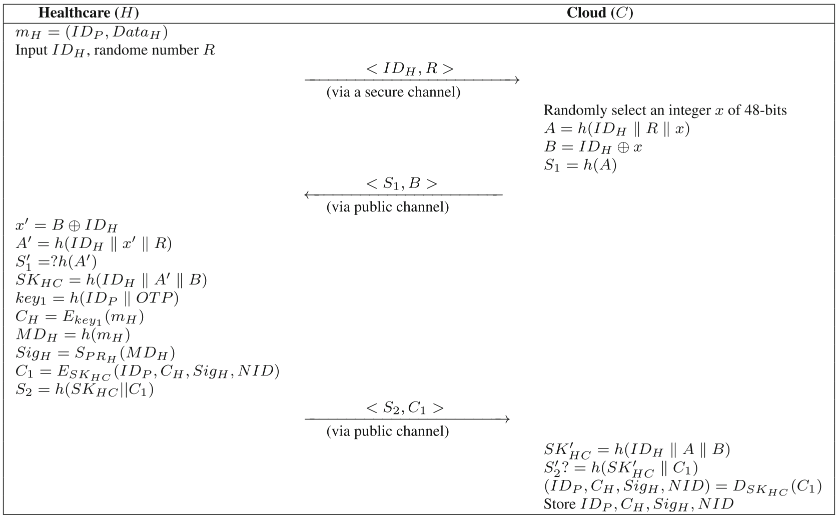
Patient data upload phase (PUP)

The patient requests the Body sensor, which is embedded in the patient's body, to collect the updated health information, and provides it to the patient via the mobile device securely. This request is made by patient by inputting his identity ID_P and password of mobile phone to the mobile device. The cloud provides an appointment sequence number sn_i, inspection report m_H to patient as shown in Fig. 3 and discussed below:

Step 1. Patient gets health information $m_B = (ID_P, Data_B)$ from body sensor via mobile phone. Then, patient inputs his identity ID_P and dynamic pseudo-random identity NID and sends it to cloud via a secure channel.

Step 2. On receiving, cloud computes $I = sn_i \oplus NID$, $S_3 = h(NID \parallel I \parallel C_H \parallel Sig_H)$ and sends $< I_3, S_3, C_H, Sig_H >$ to patient via public channel.

Step 3. Upon receiving these messages, patient computes $sn_i' = I \oplus NID$ and verifies whether $S_3' \ =?h(NID \parallel I \parallel C_H \parallel Sig_H)$ holds or not. If it does, patient authenticates cloud and computes session key between patient and cloud $SK_{PC} = h(ID_P \parallel NID)$. After performing authentication, the patient decrypts the ciphertext to obtain $m_H = D_{key_1}(C_H)$ using $key_1 = h(ID_P \parallel OTP)$. Further, the patient



Healthcare (H)

$m_H = (ID_P, Data_H)$
Input ID_H, randome number R

$< ID_H, R >$ (via a secure channel)

Cloud (C)

Randomly select an integer x of 48-bits
$A = h(ID_H \parallel R \parallel x)$
$B = ID_H \oplus x$
$S_1 = h(A)$

$< S_1, B >$ (via public channel)

$x' = B \oplus ID_H$
$A' = h(ID_H \parallel x' \parallel R)$
$S_1' =?h(A')$
$SK_{HC} = h(ID_H \parallel A' \parallel B)$
$key_1 = h(ID_P \parallel OTP)$
$C_H = E_{key_1}(m_H)$
$MD_H = h(m_H)$
$Sig_H = S_{PR_H}(MD_H)$
$C_1 = E_{SK_{HC}}(ID_P, C_H, Sig_H, NID)$
$S_2 = h(SK_{HC}||C_1)$

$< S_2, C_1 >$ (via public channel)

$SK'_{HC} = h(ID_H \parallel A \parallel B)$
$S_2'? = h(SK'_{HC} \parallel C_1)$
$(ID_P, C_H, Sig_H, NID) = D_{SK_{HC}}(C_1)$
Store ID_P, C_H, Sig_H, NID

Fig. 2 Healthcare center uploading phase (HUP)

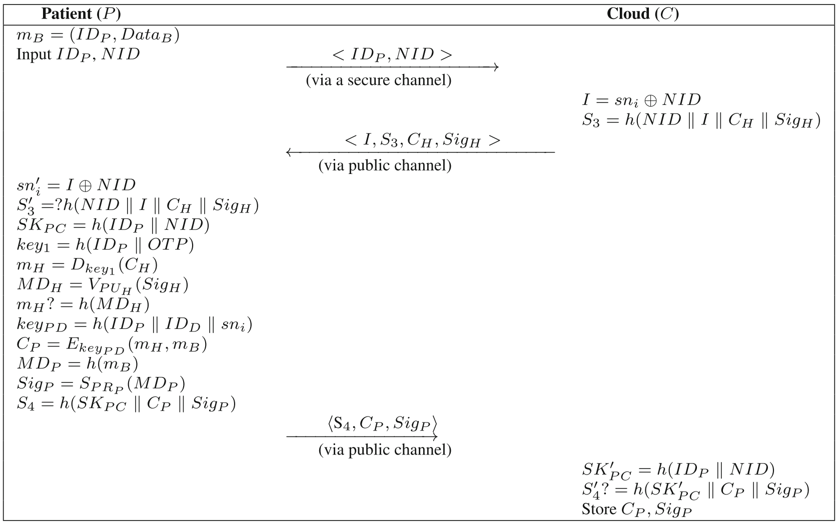

Patient (P)
Cloud (C)
$m_B = (ID_P, Data_B)$
Input ID_P, NID
$< ID_P, NID >$
(via a secure channel)
$I = sn_i \oplus NID$
$S_3 = h(NID \parallel I \parallel C_H \parallel Sig_H)$
$< I, S_3, C_H, Sig_H >$
(via public channel)
$sn'_i = I \oplus NID$
$S'_3 =?h(NID \parallel I \parallel C_H \parallel Sig_H)$
$SK_{PC} = h(ID_P \parallel NID)$
$key_1 = h(ID_P \parallel OTP)$
$m_H = D_{key_1}(C_H)$
$MD_H = V_{PU_H}(Sig_H)$
$m_H? = h(MD_H)$
$key_{PD} = h(ID_P \parallel ID_D \parallel sn_i)$
$C_P = E_{key_{PD}}(m_H, m_B)$
$MD_P = h(m_B)$
$Sig_P = S_{PR_P}(MD_P)$
$S_4 = h(SK_{PC} \parallel C_P \parallel Sig_P)$
$\langle \mathrm{S}_4, C_P, Sig_P \rangle$
(via public channel)
$SK'_{PC} = h(ID_P \parallel NID)$
$S'_4? = h(SK'_{PC} \parallel C_P \parallel Sig_P)$
Store C_P, Sig_P

Fig. 3 Patient uploading phase (PUP)

computes $MD_H = V_{PU_H}(Sig_H)$ and verifies whether $m_H? = h(MD_H)$ holds or not. If it does, patient computes key between patient and doctor $key_{PD} = h(ID_P \parallel ID_D \parallel sn_i)$ and uses the key to encrypt $C_P = E_{key_{PD}}(m_H, m_B)$. Finally, patient generates the signature corresponding to message m_B, using its private key and computes $Sig_P = S_{PR_P}(MD_P)$ where MD_P is message digest, $S_4 = h(SK_{PC} \parallel C_P \parallel Sig_P)$ and sends $< S_4, C_P, Sig_P >$ to the cloud server via public channel.

Step 4. On receiving these messages, cloud computes $SK'_{PC} = h(ID_P \parallel NID)$ and verifies whether equation $S'_4? = h(SK'_{PC} \parallel C_P \parallel Sig_P)$ holds or not. If it does, cloud stores C_P, Sig_P; otherwise, terminates the phase.

Treatment phase (TP)

Doctor performs treatment of patient by performing mutual authentication between doctor and cloud. If they are valid entities, cloud uses ID_D to find all of doctor's requests by patients, who have made appointments, and sends the patient's medical treatment data to doctor as shown in Fig. 4 and described below.

Step 1. Doctor initializes the communication by sending it's identity ID_D and a random number RD to cloud via secure channel.

Step 2. In response, cloud sends the identity ID_P of patient and sequence number sn_i via a secure channel to the doctor. The cloud computes $S_5 = h(RD \parallel Sig_P \parallel C_P)$ and sends $< S_5, Sig_P, C_P >$ to doctor via public channel.

Step 3. On receiving these messages, doctor verifies the validity of message, by checking whether $S'_5 = ?h(RD \parallel Sig_P \parallel C_P)$ holds or not. If it does, doctor authenticates the cloud and computes the session key between doctor and cloud as $SK_{DC} = h(ID_P \parallel RD \parallel sn_i)$, otherwise, rejects the message. Further, the doctor computes key_{PD} to decrypt the received messages $(m_H, m_B) = D_{key_{PD}}(C_P)$, and verifies the patient's signature using public key of patient $MD_P = V_{PU_P}(Sig)$ and checks whether $MD_P? = h(m_B)$ holds or not. If it does, doctor makes a medical diagnosis based on the reports m_H, m_B and generates medical records $m_D = (ID_P, Data_D)$, using $key_{PD} = h(ID_P \parallel ID_D \parallel sn_i)$ to encrypt messages (m_H, m_B, m_D) for generating cipher text $C_D = E_{key_{PD}}(m_H, m_B, m_D)$. Finally, the doctor signs $Sig_D = S_{PR_D}(MD_D)$ on the message m_D with its private key by computing message digest $MD_D = h(m_D)$, $S_6 = h(SK_{DC} \parallel C_D \parallel Sig_D)$ and sends $< S_6, C_D, Sig_D >$ to the cloud via public channel.

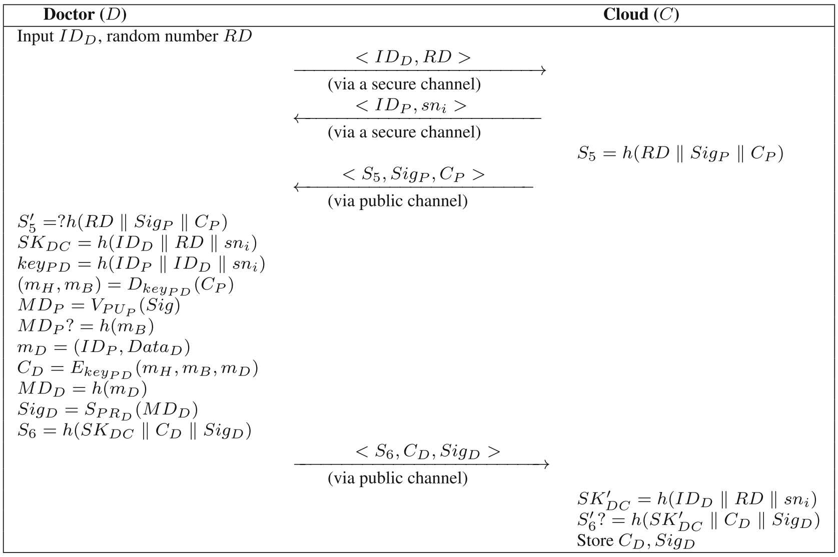
Step 4. Upon receiving these messages, cloud computes $SK'_{DC} = h(ID_P \parallel RD \parallel sn_i)$ and verifies whether $S'_6? = h(SK'_{DC} \parallel C_D \parallel Sig_D)$ holds or not. If it does, the cloud stores C_D, Sig_D otherwise, terminates and goes to Step 1.

Check up phase (CP)

After performing treatment from a doctor, the patient's report is stored in the cloud. The Patient performs authentication with the cloud and then it sends the encrypted report to patient as shown in Fig. 5. The detail description of checkup phase is as follows.

Step 1. The patient inputs its identity (ID_P), as request and sends it to cloud via secure channel.

Fig. 4 Treatment phase (TP)


Doctor (D)
Cloud (C)
Input ID_D, random number RD
$< ID_D, RD >$
(via a secure channel)
$< ID_P, sn_i >$
(via a secure channel)
$S_5 = h(RD \| Sig_P \| C_P)$
$< S_5, Sig_P, C_P >$
(via public channel)
$S'_5 =?h(RD \| Sig_P \| C_P)$
$SK_{DC} = h(ID_D \| RD \| sn_i)$
$key_{PD} = h(ID_P \| ID_D \| sn_i)$
$(m_H, m_B) = D_{key_{PD}}(C_P)$
$MD_P = V_{PU_P}(Sig)$
$MD_P? = h(m_B)$
$m_D = (ID_P, Data_D)$
$C_D = E_{key_{PD}}(m_H, m_B, m_D)$
$MD_D = h(m_D)$
$Sig_D = S_{PR_D}(MD_D)$
$S_6 = h(SK_{DC} \| C_D \| Sig_D)$
$< S_6, C_D, Sig_D >$
(via public channel)
$SK'_{DC} = h(ID_D \| RD \| sn_i)$
$S'_6? = h(SK'_{DC} \| C_D \| Sig_D)$
Store C_D, Sig_D

Step 2. On receiving, cloud computes $S_8 = h(ID_P \| C_D \| Sig_D)$ and sends $< S_8, C_D, Sig_D >$ to patient via public channel.

Step 3. On receiving these messages, patient verifies whether $S'_8 =?h(ID_P \| C_D \| Sig_D)$ holds or not. If it does, the patient decrypts the ciphertext using key_{PD} to get $(m_H, m_B, m_D) = D_{key_{PD}}(C_D)$ and verifies the signature by computing the message digest $MD_D = V_{PU_D}(Sig_D)$ and checks whether $MD_D? = h(m_D)$ holds or not. If it does, patient encrypts the messages $C_2 = E_{key_P}(m_H, m_B, m_D)$, $S_9 = h(NID \| C_2)$ and sends S_9, C_2 to the cloud server via public channel.

Step 4. Upon receiving these messages, cloud verifies whether $S'_9? = h(NID \| C_2)$ holds or not. If it does exist, the cloud stores C_2, otherwise terminates and goes to Step 1.

Security analysis

This section discusses security issues and analyzes them in our protocol. We consider an attacker $\mathcal{A}$ has the capacity to modify and eavesdrop the communicating message over the public channel. Table 2 illustrates the security comparison of proposed protocol with related existing protocols qualitatively, where 'Yes' means the respected feature is present in scheme and 'No' means not present.

Property 1 The proposed protocol provides data Confidentiality.

Proof Confidentiality is the mechanism to provide protection on transmitting of data from the adversary. Thus, encryption of data is required during transmission. The clear description for the above claim is given below.

Fig. 5 Checkup phase (CP)

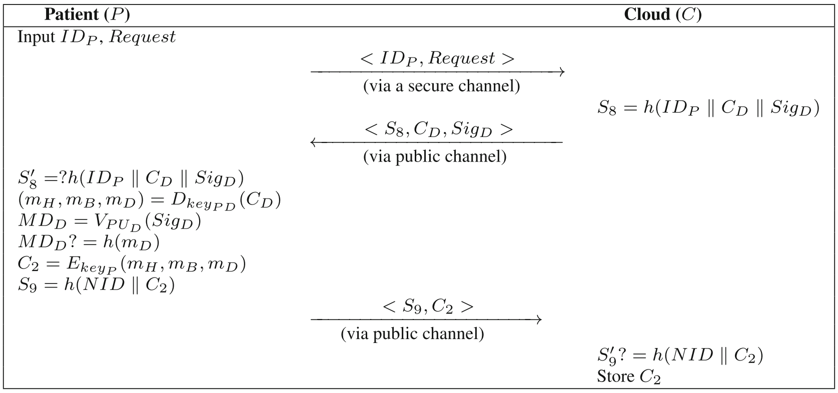


Table 2 Security analysis

Attacks	Chen et al. [9]	Chen-Yang et al. [8]	Chiou et al. [10]	Proposed protocol
Impersonation attack resistance	No	Yes	Yes	Yes
Man-in-middle attack resistance	Yes	Yes	Yes	Yes
Known-key security resistance	Yes	Yes	Yes	Yes
Data confidentiality	Yes	Yes	Yes	Yes
Non-repudiation	Yes	Yes	Yes	Yes
Known-key security resistance	Yes	Yes	Yes	Yes
Session-key security	Yes	Yes	Yes	Yes
Stolan Mobile device resistance	No	No	No	Yes
Patient anonymity	No	No	No	Yes
Message authentication	Yes	No	Yes	Yes

Yes: Attacks protected by the scheme.
No: Attacks not protected by the scheme.

- During HUP phase, the healthcare' s report m_H is encrypted with key_1 and obtains $C_H = E_{key_1}(m_H)$ to upload in cloud server.
- In PUP phase, the patient's report m_B is encrypted with key_{PD} and obtains $C_P = E_{key_{PD}}(m_H, m_B)$ to upload in the cloud.
- In TP phase, the doctor's report m_D is encrypted with key_{PD} and obtains $C_D = E_{key_{PD}}(m_H, m_B, m_D)$ to upload in the cloud.
- In CP phase, the key_P is used to encrypt $C_2 = E_{key_P}(m_H, m_B, m_D)$.

Hence, if the adversary $\mathcal{A}$ tries to obtain information during communication, he gets encrypted data which can't be decrypted without the key. Thus, our scheme supports confidentiality. □

Property 2 The proposed protocol provides the Non-repudiation.

Proof Non-repudiation states as the ability to ensure that a party to a contract or a communication cannot deny the authenticity of their signature on a document or the sending of a message that they originated. During HUP phase, the healthcare signs a message $Sig_H = S_{PR_H}(MD_H)$ which is verified by patient $m_H? = h(MD_H)$ by computing $MD_H = V_{PU_H}(Sig_H)$. The signature Sig_H ensures that the report is issued from legal healthcare center, and only the patient can perform the verification of Sig_H. If the healthcare records have some problems, the healthcare cannot be denied from the fact that the message is sent by healthcare. Hence, non-repudiation is ensured in our protocol. □

Property 3 The proposed protocol provides Message authentication.

Proof Message authentication [22] is a mechanism used to verify the integrity of the message. Here, we describe message authentication in each phase:

- During HUP phase, healthcare center receives S_1, B and verifies its validity. Similarly, the cloud receives S_2, C_1 and verifies it using SK_{HC} as $S_2? = h(SK_{HC}||C_1)$. If an adversary $\mathcal{A}$ tries to alter any value of the message the cloud server will recognize it.
- In PUP phase, the patient verifies I, S_3, C_H, Sig_H as $S_3 =?h(NID \parallel I \parallel C_H \parallel Sig_H)$ and the cloud verifies received S_4, C_P, Sig_P as $S_4' = h(SK_{PC}' \parallel C_P \parallel Sig_P)$, Hence, message authentication between patient and cloud is verified.
- In TP phase, doctor verifies received S_5, Sig_P, C_P using SK_{DC} as $S_5? = h(SK_{DC} \parallel C_P \parallel Sig_P)$ and the cloud verifies the received message S_6, Sig_D, C_D as $S_6? = h(SK_{DC}' \parallel C_D \parallel Sig_D)$ and if any of the verification fails, the messages will not be accepted.
- In CP phase, the patient verifies S_8, C_D, Sig_D as $S_8? = h(ID_P \parallel C_D \parallel Sig_D)$ and the cloud verifies S_9, C_2 as $S_9? = h(NID \parallel C_2)$.

Thus, our scheme protects message authentication in every phase. □

Property 4 The Patient anonymity of the proposed protocol is secure.

Proof Our scheme can preserve anonymity property by hiding original identity. During HUP phase, ID_P is encrypted with SK_{HC} as $C_1 = E_{SK_{HC}}(ID_P, C_H, Sig_H, NID)$ and only can be decrypted by cloud having session key $(ID_P, C_H, Sig_H, NID) = D_{SK_{HC}}(C_1)$ to obtain ID_P, C_H, Sig_H, NID. The identity of patient ID_P cannot be derived without the knowledge of SK_{HC}. Furthermore, SK_{HC} cannot be derived due to one-way hash function. Therefore, our protocol provides patient anonymity. □

Property 5 The Man-in-the-middle attack does not exist in our scheme.

Proof In our scheme, the patient, healthcare center and doctor use encryption of data before sending message over public channel and verify message before accepting it. If the message is altered during transmission it fails the verification process. Thus, the attacker will not get success in the alteration process. Therefore, our scheme protects the man-in-middle attack. □

Property 6 The proposed protocol provides protection against the known-key security.

Proof In our scheme, the patient, healthcare center, the doctor and cloud can use random numbers to generate session key. Even if an attacker steals previous session key, he/she cannot generate the session key for the next times. Therefore, our protocol is protected against known-key security. □

Property 7 The proposed protocol protects the stolen mobile device attack.

Proof Suppose that an adversary steals the mobile device of a legal patient, the adversary $\mathcal{A}$ cannot obtain any of the secret information of the patient. As the mobile device receives $m_B = (ID_P, Data_B)$, which is accessible only by inputting valid identity of patient (ID_P) and valid password of mobile device, which is only known to the valid patient. Therefore, the attacker is not able to break the system even if he/she gets the mobile device of the registered patients. □

Property 8 The proposed protocol protects the impersonation attack.

Proof If an adversary $\mathcal{A}$ interrupts in between communication, the adversary can trap the transmitting messages, which is transferred via the public channel. After getting the transmitted message the adversary $\mathcal{A}$ can alter the message, and re-transmit the modified. Moreover, the modified message has to pass the verification process performed by the other party, which is impossible in the proposed protocol. The detail is described in terms of healthcare center update phase. However, the concept is same in other phases.

- An adversary $\mathcal{A}$ tries to impersonate as a legal cloud, and eavesdrops the transmitted message $< S_1, B >$ and tries to re-compute B using ID_H, which is unknown parameter and only known to the authenticated parties. Also the computation of S_1 is not possible, as it involves hashing of A. If the adversary $\mathcal{A}$ tries to impersonate as healthcare center by using different identity or guessing the ID_H. It results in computing the value of A as it involves two other parameters, 1) secret key x of cloud 2) Random number R generated by the healthcare center. Note that, guessing of all the value at the same time is not possible due to preserving high entropy property. If adversary $\mathcal{A}$ uses incorrect value to compute S_1, B, the verification process will not pass. Hence, computation of S_1 depends on A and the computation of A depend on x. The incorrect value of x leads to incorrect value of S_1. Thus, the adversary cannot impersonate a legal cloud.
- If the adversary$\mathcal{A}$ tries to impersonate as a valid healthcare center, he first eavesdrops the transmitted message $< S_2, C_1 >$ from the public channel and re-constructs the new message. The computation of both messages involve the session key between the healthcare center and cloud, which involves hash operation in computation. Thus, the adversary cannot impersonate the healthcare.

From the above points it is clear that the proposed protocol is protected against impersonation attack. □

Property 9 The proposed protocol is secure against the session-key security.

Proof The session key security is one of the very important parameters which we have considered in order to design our protocol. It is necessary that the session key is only known to the legitimate parties. In this protocol, there are total three session keys which are computed between 1) Healthcare center and Cloud, 2) Patient and Cloud and 3) Doctor and Cloud. All of these session keys are well secured. Here we describe the security in terms of healthcare center update phase. However, the concept is same in other phases.

The session key between Healthcare center- Cloud $SK_{HC} = h(ID_H \parallel A \parallel B)$ comprises hashing of ID_H, A, B that need to be determined by the attacker $\mathcal{A}$ for generating an exact session key. Property 8 shows that the adversary $\mathcal{A}$ can not extract the parameter A, B from the communicating messages between helthcare and cloud. Similarly, the identity of hospital ID_H is sent via a secure channel to cloud, attacker can not access the identity ID_H. Without knowing the parameters ID_H, A, B an adversary $\mathcal{A}$ cannot compute the session key SK_{HC}. Thus, the session key can only be generated by a legitimate party. □

Table 3 Computation cost of our protocol with related schemes

Schemes	Chen et al. [9]	Chen-Yang et al. [8]	Chiou et al. [10]	Proposed protocol
HUP	$1T_{Sig} + 1T_M + 2T_P + 4T_S + 2T_H + 3T_A$	$T_{Sig} + 4T_M + 4T_P + 2T_S + 6T_H + 1T_A$	$T_{Sig} + 3T_P + 2T_S + 7T_H$	$1T_{Sig} + 3T_S + 11T_H$
PUP	$1T_M + 2T_P + 4T_S + 2T_H + 3T_A$	$T_{Sig} + 4T_M + 4T_P + 3T_S + 6T_H + 1T_A$	$T_{Sig} + 4T_P + 2T_S + 12T_H$	$2T_{Sig} + 2T_S + 10T_H$
TP	$2T_{Sig} + 1T_M + 2T_P + 7T_S + 2T_H + 4T_A$	$2T_{Sig} + 4T_M + 4T_P + 4T_S + 6T_H$	$2T_{Sig} + 4T_M + 4T_P + 4T_S + 6T_H$	$2T_{Sig} + 2T_S + 9T_H$
CP	N/A	N/A	$T_{Sig} + 2T_P + 2T_S + 8T_H$	$1T_{Sig} + 2T_S + 5T_H$
EP	N/A	$2T_{Sig} + 3T_P + 6T_S + 4T_H$	N/A	N/A
Total cost	$3T_{Sig} + 3T_M + 6T_P + 15T_S + 6T_H + 10T_A \approx 4.7091 sec$	$6T_{Sig} + 12T_M + 15T_P + 15T_S + 22T_H + 2T_A \approx 4.379 sec$	$5T_{Sig} + 4T_M + 13T_P + 10T_S + 33T_H \approx 2.7705 sec$	$6T_{Sig} + 9T_S + 35T_H \approx 2.086 sec$

Performance analysis

In this section, we evaluate the performance of our protocol with the related protocols used in cloud for secure medical data exchange, such as Chen et al. [9], Chiou et al. [10] and Chen-Yang et al. [8] protocols. The comparison is performed in all the phases of protocol like HUP, PUP, TP, CP and EP. Chuang et al. [8] uses the emergency phase (EP) in his scheme.

Table 3 summarizes the computation cost of our protocol with related protocols. It is well known that the computational cost of XOR ($\oplus$) and concatenation ($||$) operations are considered as negligible compared to other operations such as symmetric encryption, multiplication, bilinear pairing, etc. As it is clear from Table 3 that the proposed protocol has less computation cost than the existing protocols used in cloud for medical exchange of data. Hence, our scheme is light weighted.

We have used several crypto-operations in this article based on the information available in Ref. [10] to evaluate computation cost of our protocol as well as existing related research. In Ref. [10], Android phone and Windows 7 OS is used and the system configurations of mobile phone is Android 4.4.4KTU84P with a 1.8 GHz processor and 2GB RAM. The configurations of computer system is Windows 7, Professional with an Intel (R) core (TM) 2 Quad CPU Q8300, @2.50Hz, and 2GB RAM. The execution time in second for the different time complexity notations are as follows:

T_{Sig}: the time for computing executing/verifying a signature ($T_{Sig} \approx 0.3317 sec.$)

T_A: the time for computing asymmetric encryption or decryption operation ($T_A \approx 0.3057 sec.$)

T_M: the time for computing multiplication ($T_M \approx 0.0503 sec.$)

T_P: the time for computing a bilinear pairing operation ($T_P \approx 0.0621 sec.$)

T_S: the time for computing symmetric encryption or decryption operation ($T_S \approx 0.0087 sec.$)

T_H: the time for computing one-way hash function ($T_H \approx 0.0005 sec.$)

We summarize the communication cost in Table 4 of our protocol with related protocols. Assume the bit length of identity, timestamp and randomly generated numbers to be 48-bits each, the bilinear pairing and cryptographic hash function as 160 bits, the length of symmetric cryptosystem,

Table 4 Communication cost of our protocol with related protocol

Communication cost in *bits*				
Phases	Chen et al. [9]	Chen-Yang et al. [8]	Chiou et al. [10]	Proposed protocol
HUP	816	1936	704	592
PUP	816	2064	1600	1744
TP	944	2192	2112	1792
CP	N/A	N/A	2112	1184
EP	N/A	1760	N/A	N/A
Total Cost(in bits)	2576	7952	6528	5312

asymmetric- algorithm/signature to be 128, 512-bits respectively. The proposed scheme is efficient in terms of communication cost as its communication cost is 5312-bits, which is less than the [8, 10] having costs of 7952, 6528-bits respectively. Although, it is greater than [9] having cost of 2576-bits. But [9] is vulnerable to impersonation attack, user anonymity and stolen mobile device attack, and hence, not suitable for practical implementation in cloud. However, our scheme can resist all the above attacks.

Conclusion

We have outlined Chiou et al.'s authentication scheme designed for a TMIS using cloud environment. On cryptanalysis, it is found that the scheme is vulnerable to patient anonymity, mobile device stolen attack. We have then proposed a new user authentication and session key agreement scheme for the same, which fixed the mentioned security weaknesses. In addition, we showed that the proposed scheme provides better security than other existing schemes through the security analysis. Our protocol is also efficient in terms of performance such as computation and computation overheads. We have not considered some security issues of the cloud environment in this work. Hence, it is our future research to solve the above mentioned problems.

Acknowledgments The authors extend their sincere appreciations to the Deanship of Scientific Research at King Saud University for its funding this Prolific Research Group (PRG-1436-16).

References

1. Amin, R., Cryptanalysis and efficient dynamic id based remote user authentication scheme in multi-server environment using smart card. *IJ Netw. Secur.* 18(1):172–181, 2016.
2. Amin, R., and Biswas, G. P., Cryptanalysis and design of a three-party authenticated key exchange protocol using smart card. *Arab. J. Sci. Eng.* 40(11):3135–3149, 2015.
3. Amin, R., and Biswas, G. P., A secure three-factor user authentication and key agreement protocol for tmis with user anonymity. *J. Med. Syst.* 39(8):1–19, 2015.
4. Amin, R., Islam, S. H., Biswas, G. P., Khan, M. K., and Kumar, N.: A robust and anonymous patient monitoring system using wireless medical sensor networks. Future Generation Computer Systems. doi:10.1016/j.future.2016.05.032, 2016.
5. Amin, R., Islam, S. H., Biswas, G. P., Khan, M. K., and Li, X., Cryptanalysis and enhancement of anonymity preserving remote user mutual authentication and session key agreement scheme for e-health care systems. *J. Med. Syst.* 39(11):140, 2015.
6. Chaudhry, S. A., Khan, M. T., Khan, M. K., and Shon, T., A multiserver biometric authentication scheme for tmis using elliptic curve cryptography. *J. Med. Syst.* 40(11):230, 2016.
7. Chaudhry, S. A., Naqvi, H., Shon, T., Sher, M., and Farash, M. S., Cryptanalysis and improvement of an improved two factor authentication protocol for telecare medical information systems. *J. Med. Syst.* 39(6):65–75, 2015.
8. Chen, C. L., Yang, T. T., Chiang, M. L., and Shih, T. F., A privacy authentication scheme based on cloud for medical environment. *J. Med. Syst.* 38(11):1–16, 2014.
9. Chen, C. L., Yang, T. T., and Shih, T. F., A secure medical data exchange protocol based on cloud environment. *J. Med. Syst.* 38(9):1–12, 2014.
10. Chiou, S. Y., Ying, Z., and Liu, J., Improvement of a privacy authentication scheme based on cloud for medical environment. *J. Med. Syst.* 40(4):1–15, 2016.
11. Debiao, H., Jianhua, C., and Rui, Z., A more secure authentication scheme for telecare medicine information systems. *J. Med. Syst.* 36(3):1989–1995, 2012.
12. Giri, D., Sherratt, R.S., Maitra, T., and Amin, R., Efficient biometric and password based mutual authentication for consumer usb mass storage devices. *IEEE Trans. Consum. Electron.* 61(4):491–499, 2015.
13. Gope, P., and Amin, R., A novel reference security model with the situation based access policy for accessing ephr data. *J. Med. Syst.* 40(11):41–53, 2016.
14. Gope, P., and Hwang, T., A realistic lightweight anonymous authentication protocol for securing real-time application data access in wireless sensor networks. *IEEE Trans. Ind. Electron.* 63(11):7124–7132, 2016.
15. He, D., Kumar, N., Shen, H., and Lee, J. H., One-to-many authentication for access control in mobile pay-tv systems. *Sci. China Inf. Sci.* 59(5):1–14, 2015.
16. He, D., Kumar, N., Wang, H., Wang, L., Choo, K. K. R., and Vinel, A.: A provably-secure cross-domain handshake scheme with symptoms-matching for mobile healthcare social network. IEEE Transactions on Dependable and Secure Computing (99), 1–1. doi:10.1109/TDSC.2016.2596286, 2016.
17. He, D., and Wang, D., Robust biometrics-based authentication scheme for multiserver environment. *IEEE Syst. J.* 9(3):816–823, 2015.
18. He, D., Zeadally, S., Kumar, N., and Lee, J. H.: Anonymous authentication for wireless body area networks with provable security (99) 1–12. doi:10.1109/JSYST.2016.2544805, 2016.
19. Islam, S., Obaidat, M. S., and Amin, R.: An anonymous and provably secure authentication scheme for mobile user. International Journal of Communication Systems. doi:10.1002/dac.3126, 2016.
20. Islam, S. H., Amin, R., Biswas, G. P., Farash, M. S., Li, X., and Kumari, S.: An improved three party authenticated key exchange protocol using hash function and elliptic curve cryptography for mobile-commerce environments Journal of King Saud University-Computer and Information Sciences. doi:10.1016/j.jksuci.2015.08.002, 2015.
21. Jiang, Q., Ma, J., Ma, Z., and Li, G., A privacy enhanced authentication scheme for telecare medical information systems. *J. Med. Syst.* 37(1):1–8, 2013.
22. Karati, A., Amin, R., and Biswas, G. P., Provably secure threshold-based abe scheme without bilinear map. *Arab. J. Sci. Eng.* 41(8):3201–3213, 2016.
23. Khan, M. K., and Kumari, S., An authentication scheme for secure access to healthcare services. *J. Med. Syst.* 37(4):1–12, 2013.
24. Kumar, R., Amin, R., Karati, A., and Biswas, G. P., Secure remote login scheme with password and smart card update facilities. In: Proceedings of the 4th international conference on frontiers in intelligent computing: Theory and applications (FICTA) 2015, pp. 495–505: Springer, 2016.

25. Kumari, S., Khan, M. K., and Kumar, R., Cryptanalysis and improvement of ’a privacy enhanced scheme for telecare medical information systems’. *J. Med. Syst.* 37(4):1–11, 2013.
26. Lee, C. C., Hsu, C. W., Lai, Y. M., and Vasilakos, A., An enhanced mobile-healthcare emergency system based on extended chaotic maps. *J. Med. Syst.* 37(5):1–12, 2013.
27. Li, C. T., Lee, C. C., and Weng, C. Y., A secure chaotic maps and smart cards based password authentication and key agreement scheme with user anonymity for telecare medicine information systems. *J. Med. Syst.* 38(9):1–11, 2014.
28. Li, X., Kumari, S., Shen, J., Wu, F., Chen, C., and Islam, S. H.: Secure data access and sharing scheme for cloud storage. Wireless Personal Communications pp.1–20. doi:10.1007/s11277-016-3742-6, 2016.
29. Li, X., Niu, J., Karuppiah, M., Kumari, S., and Wu, F., Secure and efficient two-factor user authentication scheme with user anonymity for network based e-health care applications. *J. Med. Syst.* 40(12):267–277, 2016.
30. Li, X., Niu, J., Khan, M. K., and Liao, J., An enhanced smart card based remote user password authentication scheme. *J. Netw. Comput. Appl.* 36(5):1365–1371, 2013.
31. Maitra, T., Obaidat, M. S., Amin, R., Islam, S., Chaudhry, S. A., and Giri, D.: A robust elgamal-based password-authentication protocol using smart card for client-server communication International Journal of Communication Systems. doi:10.1002/dac.3242, 2016.
32. Mishra, D., Mukhopadhyay, S., Chaturvedi, A., Kumari, S., and Khan, M. K., Cryptanalysis and improvement of yan others.’s biometric-based authentication scheme for telecare medicine information systems. *J. Med. Syst.* 38(6):1–12, 2014.
33. Mishra, D., Srinivas, J., and Mukhopadhyay, S., A secure and efficient chaotic map-based authenticated key agreement scheme for telecare medicine information systems. *J. Med. Syst.* 38(10):1–10, 2014.
34. Shao, J., Lin, X., Lu, R., and Zuo, C., A threshold anonymous authentication protocol for vanets. *IEEE Trans. Veh. Technol.* 65(3):1711–1720, 2016.
35. Sureshkumar, V., Anitha, R., Rajamanickam, N., and Amin, R.: A lightweight two-gateway based payment protocol ensuring accountability and unlinkable anonymity with dynamic identity Computers & Electrical Engineering. doi:10.1016/j.compeleceng.2016.07.014, 2016.
36. Sutrala, A. K., Das, A. K., Odelu, V., Wazid, M., and Kumari, S., Secure anonymity-preserving password-based user authentication and session key agreement scheme for telecare medicine information systems. *Comput. Methods Prog. Biomed.* 135:167–185, 2016.
37. Tan, Z., An efficient biometrics-based authentication scheme for telecare medicine information systems. *Network* 2(3):200–204, 2013.
38. Wazid, M., Das, A. K., Kumari, S., Li, X., and Wu, F., Design of an efficient and provably secure anonymity preserving three-factor user authentication and key agreement scheme for tmis. *Secur. Commun. Netw.* 9(13):1983–2001, 2016.
39. Wei, J., Hu, X., and Liu, W., An improved authentication scheme for telecare medicine information systems. *J. Med. Syst.* 36(6):3597–3604, 2012.
40. Wu, Z. Y., Chung, Y., Lai, F., and Chen, T. S., A password-based user authentication scheme for the integrated epr information system. *J. Med. Syst.* 36(2):631–638, 2012.
41. Wu, Z. Y., Lee, Y. C., Lai, F., Lee, H. C., and Chung, Y., A secure authentication scheme for telecare medicine information systems. *J. Med. Syst.* 36(3):1529–1535, 2012.
42. Wu, Z. Y., Tseng, Y. J., Chung, Y., Chen, Y. C., and Lai, F., A reliable user authentication and key agreement scheme for web-based hospital-acquired infection surveillance information system. *J. Med. Syst.* 36(4):2547–2555, 2012.
43. Yan, X., Li, W., Li, P., Wang, J., Hao, X., and Gong, P., A secure biometrics-based authentication scheme for telecare medicine information systems. *J. Med. Syst.* 35(5):1–6, 2013.
44. Yang, H., Kim, H., and Mtonga, K., An efficient privacy-preserving authentication scheme with adaptive key evolution in remote health monitoring system. *Peer-to-Peer Netw. Appl.* 8(6):1059–1069, 2015.
45. Zhu, Z., An efficient authentication scheme for telecare medicine information systems. *J. Med. Syst.* 36(6):3833–3838, 2012.