Journal of Biomedical Informatics 78 (2018) 24–32

Contents lists available at ScienceDirect

Journal of Biomedical Informatics

journal homepage: www.elsevier.com/locate/yjbin

Practical and secure telemedicine systems for user mobility

Fatemeh Rezaeibagha*, Yi Mu*

Institute of Cybersecurity and Cryptology, School of Computing and Information Technology, University of Wollongong, NSW 2522, Australia

ARTICLE INFO

Keywords:
Telemedicine
Security
Authentication
Anonymity
Mobility

ABSTRACT

The application of wireless devices has led to a significant improvement in the quality delivery of care in telemedicine systems. Patients who live in a remote area are able to communicate with the healthcare provider and benefit from the doctor consultations. However, it has been a challenge to provide a secure telemedicine system, which captures users (patients and doctors) mobility and patient privacy. In this work, we present several secure protocols for telemedicine systems, which ensure the secure communication between patients and doctors who are located in different geographical locations. Our protocols are the first of this kind featured with confidentiality of patient information, mutual authentication, patient anonymity, data integrity, freshness of communication, and mobility. Our protocols are based on symmetric-key schemes and capture all desirable security requirements in order to better serve our objectives of research for secure telemedicine services; therefore, they are very efficient in implementation. A comparison with related works shows that our work contributes first comprehensive solution to capture user mobility and patient privacy for telemedicine systems.

1. Introduction

In traditional medical systems, patients have to visit a clinic or a hospital to have a doctor consultation and treatment; therefore, it is inconvenient for elderly patients and patients residing in rural and distant areas, especially for those with chronic diseases. Thanks to Telecare Medical Information Systems (TMIS) which have made outstanding advances and provided efficient communications among patients and doctors. Although many countries are demanding home-based long-term care in order to provide efficient treatment for patients and increase their quality of lives, it also raises some security and privacy concerns among healthcare providers and users in terms of disclosing patient information in insecure communication channels [1].

The growth of wireless network devices in medical services has been in an inevitable pace to offer various applications in healthcare. There are many studies addressing the security pitfalls of TMIS through various methods; however, there are still many security issues and implementation obstacles. Medical information should be protected through authenticated channels and cryptographic solutions. There are many works in the literature, which have proposed various solutions. However, there are still some open questions for various security needs in telemedicine applications. It is in fact a challenge to provide an efficient and secure system which captures all necessary needs for remote medical services.

1.1. Motivation and contribution

In telemedicine applications, the established services are usually considered for patients and doctors who reside in some fixed locations, where the telemedicine infrastructure can be supplied. However, in practice, patients and doctors can mobile, in a sense that they might need to travel to a different location. Therefore, many new security issues will arise due to changing the computing environment. Notice that these issues have not been fully addressed in the literature. The motivation of this work is to provide sound security solutions to the mobility of patients and doctors by achieving both security and privacy. With a distinct feature of our work, our solution provides identity privacy against the disclosure of patient identities.

The contribution of our work is outlined as follows. In this work, for the first time, we capture all the aforementioned features for a practical and ideal telemedicine service. As the main contribution, we systematically studied how to secure the wireless medical systems while patients and doctors can be mobile, and in the meanwhile, patients can still remain anonymous. We overcome the security hurdle due to the mobility of patients and doctors. We present our approaches with four application scenarios in terms of the mobility of users, and show that in those scenarios, patients and doctors can establish a secure communication channel which meets all our security and privacy requirements, i.e., patient's privacy (anonymity), communication freshness, data confidentiality, and data integrity.

* Corresponding authors.
E-mail addresses: fr683@uowmail.edu.au (F. Rezaeibagha), ymu@uow.edu.au (Y. Mu).

https://doi.org/10.1016/j.jbi.2017.12.011
Received 8 August 2017; Received in revised form 2 December 2017; Accepted 20 December 2017
Available online 27 December 2017
1532-0464/

1.2. Related work

Before presenting our protocols, we review related work and show the research gap that our work will fill.

In the literature, many security schemes have been proposed to ensure secure communication. These studies employed user authentication and session key agreement protocols [2,3]. The work of [2] is based on smart cards, while the work of [3] does not require any smart cards. Although it was claimed a number of security features including anonymity in these papers, the proposed solutions suffer from different types of attacks, including insider attacks, anonymity problems, replay attacks, etc. Also the works presented by [4–9] do not handle user mobility issues.

In [4], authors proposed an authentication protocol based on discrete logarithm and hashing. The protocol allows two parties to authenticate each other, but it is unsuitable for a distributed environment and user mobility. They did not consider anonymity in their work. This work was later found flawed [6,7]. The work in [6,7] fixed the flaw in [4]. There are further proposed solutions for telemedicine systems offering authentication schemes [5–8], which provide authentications services for patients, healthcare server, doctors, along with the security analysis on impersonation attack, replay attack, message authenticity, backward/forward secrecy, and confidentiality. These work do not consider user mobility, although [7] has considered anonymity.

We noticed that there are some works which have considered confidentiality and authentication in telemedicine (or can be potentially applied to telemedicine). For instance, Yang et al. [10] proposed a privacy-preserving authentication scheme with adaptive key evolution. A similar work proposed by Chen et al. [11] discussed multi-channel safety authentication protocols in wireless networks. Jiang et al. [12] also proposed an authentication scheme for wireless body area networks in mhealth. However, they do not address the mobility of patients and doctors.

In addition, some other studies provide biometric-based authentication schemes [9], where gateways were designated to register patients and doctors, which can threat the security if an attacker eavesdrops the intermediate point. Rahman et al. [13] also considered patient privacy using an attribute-based setting.

We compare the security properties of our protocols with some other proposed solutions in Table 1. As it is clearly stated, aforementioned studies in the literature offer some security properties which have been mentioned; however, they have only captured some basic security properties without considering mobility. Our work is the only work which captures this important feature for telemedicine.

Note that all schemes we have mentioned are public-key based, therefore they introduce substantial computational overhead to the system. Moreover, none of these schemes are catered for secure user mobility. Since our protocols are symmetric-key based, they are much more efficient in applications and are designed for user mobility and privacy.

In addition, there are some other works related to secure telemedicine in the literature. As examples, we listed a few. Chatterjee et al. [14] have proposed a fine grained access control. In [15], the authores proposed a password based authentication and key establishment protocol for telemedicine applications, where the security of the system and the anonymity are based on a user password. In [16], the authors introduced a three-factor authentication scheme with user anonymity. There are some works in the literature about distributed telemedicine applications (e.g., [17,18]). We note that none of these works has considered user mobility.

In the following sections, we first present our system model and our security assumptions. Based on them we demonstrate four practical telemedicine scenarios and their secure communication protocols, along with two clinical examples. We also provide the security analysis.

Table 1
Comparison of the proposed scheme with related works.

Security properties	[2]	[3]	[5]	[6]	[7]	[10]	[12]	Our protocol
Confidentiality	Yes	Yes	Yes	No	Yes	Yes	Yes	Yes
Authentication	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Anonymity	No	No	No	No	Yes	No	Yes	Yes
Mobility	No	No	No	No	No	No	No	Yes

2. System model and security requirements

2.1. System model

We consider a general case in a medical system, which consists of a healthcare center, patients, and doctors. Consider a scenario of telemedicine, it consists of a healthcare provider, authentication server, patients, and doctors. For simplicity and clearness of presentation, we refer to “authentication server” as an entity which represents healthcare provider, system administration and security service. Patients and doctors are registered in the authentication server.

We provide a solution to telemedicine which can be wireless and wired. Our solution captures security and privacy requirements mentioned earlier along with mobility of users. In Fig. 1, we illustrate our system model, where the healthcare center provides the entire service, users (doctors and patients) are located at different regions or domains, which are serviced by authentication servers (local or remote), depending on the user location. In our model, we allow users to move to a different domain dynamically while maintaining their ability to communicate securely with each other. Our system captures all possible situations and provides the same level of security services to users, regardless of the type of location.

Note: Actually, we are not trying to invent a new communication infrastructure to cater for our purposes; instead, we are trying to utilize the existing communication infrastructure with a minimal modification. Taking the phone based services as an example, our system could be built on top of these existing telecommunication servers and provides some additional functionalities to carter for our needs. A centralized design could be seen as one of the choices. However, notice that in a large distributed environment, a centralized service has both pros and cons. The obvious con for a centralized server could be a potential bottleneck, which might attract packet flooding attacks. To avoid overloading the centralized server, we chose the distributed design. Actually, this practice has been widely adopted in distributed environments, for example, Kerberos.

2.2. Threat model and notations

We presume that users (patients and doctors) belong to their home-area network, in which they are registered. Users share a long-term secret key with their home server. The setup of the key is done at the registration. Our system only requires symmetric keys, which offer much better computational efficiency. We consider the following security assumptions for our protocols:

We assume that all users share a long-term key with their home server (or authentication server). With this as the basis, we can construct secure channels against eavesdropping from outsiders. Here, by outsiders, we mean that anyone who is not registered with the system and those who have registered with the system but are not involved in the protocol execution.

Definition 1 (*Confidentiality*). Assuming that the underlying symmetric-key scheme used in our protocols is secure, the messages transmitted between users are then secured against all outsiders.

We require all patients’ identities to be protected against outsiders. Notice that all user identities transmitted in our protocols are encrypted with the corresponding key.

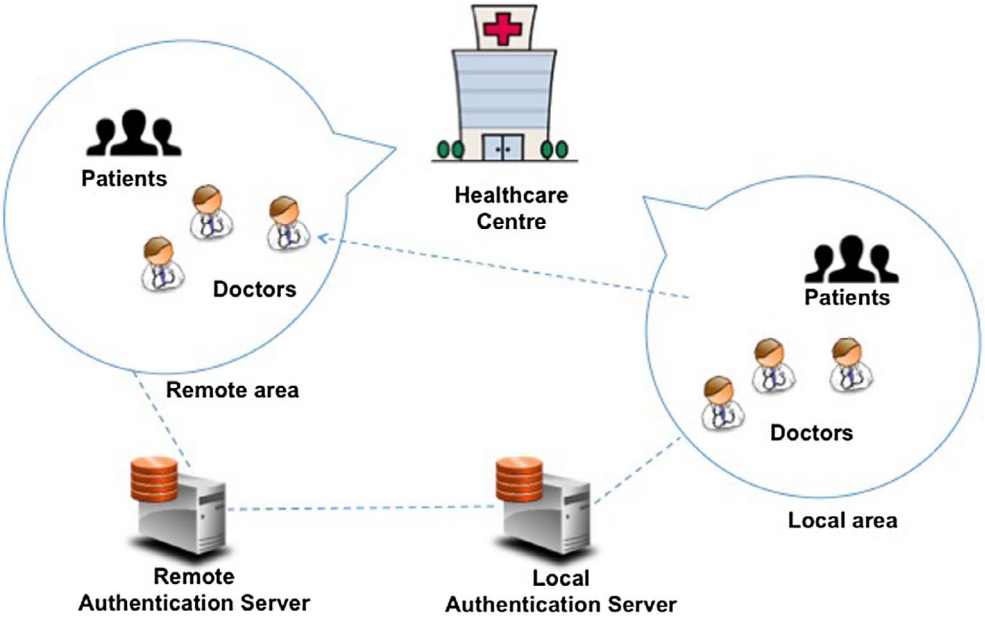

Fig. 1. System model.

Definition 2 (*Anonymity*). Assuming that the underlying symmetric-key scheme used in our protocols is secure, the identities of patients are protected against all outsiders.

We require mutual authentication between patient and servers as well as patient and doctor. We achieve mutual authentication by allowing trust between users and their home server as well as trust amongst all servers.

Definition 3 (*Authentication*). Assuming the underlying MAC scheme is secure and assuming the trust between users and their home server and the trust amongst all servers, the mutual authentication amongst all parties involved in a protocol execution is ensured.

Freshness against replay attacks from outsiders. We achieve this feature by using nonces. Notice that we can add timestamps to our protocols to secure them against suppress attacks; however, for simplicity we omit it. We only use the timestamp for service tickets.

Definition 4 (*Freshness*). Assuming that the underlying symmetric-key scheme used in our protocols is secure, our protocols are secure against reply attacks from all outsiders.

Integrity against all outsiders is achieved with message authentication code (MAC).

Definition 5 (*Integrity*). Assuming the underlying MAC scheme is secure, the messages transmitted in a protocol execution cannot be tampered by any outsider.

Table 2
Notations.

U_i: the i-th user's ID (a user could be a patient P_i or a doctor D_i)
$\mathsf{U}^s_{i,j}$: the i-th user's j-th subliminal ID
HS: Home-server
RS: Remote-server
$K_{u,v}$: Shared secret-symmetric key between party u and v
k_s: Secret session key
T: Time stamp
$[\mathsf{Data}]_k$: Data encrypted with a symmetric key k
$\mathsf{h}(\cdot)$: A secure cryptographic hash function
n_u: Nonce generated by user U
$\mathsf{U}_i \rightarrow \mathsf{U}_j$: message: U_i sends message to U_j

In Table 2, we provide the notations used in the description of the protocols.

2.3. System setup

There are two types of servers in our system model, including home server and remote server.

- **Home Server** (HS): HS is situated in the home-area network that stores the real identity of user U, who can be a doctor (D) or a patient (P), and a long-term secret symmetric key $K_{u,hs}$, which is shared by a user and HS. HS also maintains a routing table (Tab_{hs}) which stores the real identity of the user and the corresponding subliminal identity and provides the mapping of them. It also stores a long-term secret key $K_{hs,rs}$ that is used to communicate between the home server HS and the remote server RS.
- **Remote Server** (RS): RS is situated in the remote-area network and stores a long-term secret key $K_{hs,rs}$ that is used to communicate between HS and RS and a long-term secret key $K_{ru,rs}$, which is shared by the remote user and RS. To remote users, RS is their HS, if they have registered at the RS as their home server. Similarly to the HS, the RS maintains a routing table (Tab_{rs}) to service its registered users.
  All patients and doctor need to register with their home server in order to receive the service.
- **Patient Registration**: The registration phase is a one-time process between a patient and its home server HS. The purpose of patient registration is to setup the shared long-term secret key and register its identity. In turn, the patient will receive a subliminal identity and the secret key $K_{p,hs}$. As an option, $K_{p,hs}$ can be derived from the patient's password. In this case, the patient should setup its password during the registration phase. The HS will update its table which stores the mappings of real identity and subliminal identity. A subliminal identity could be an IP number or a phone number, which is randomly generated and unused in the current system, depending on the type of the system.
- **Doctor Registration**: Only difference between patient registration and doctor registration is that the doctor is not anonymous, i.e., there is no need to assign a subliminal identity to a doctor. Therefore, the doctor identity is registered with its HS and obtains a long-term secret key shared with the HS. As an option, $K_{d,hs}$ can be

derived from the doctor's password. In this case, the doctor should setup its password during the registration phase.

Note: We do not assume anonymity of doctors, as in common practice, doctors' names are known to the public. Our protocols have considered the rigorous security design to protect doctors' security and privacy, in that all doctors are required to register with a system and are assigned with a secret key shared by the corresponding doctor and his home server which is assumed to be trusted. To be a doctor, a user must have a valid secret key, as all communication flows are encrypted with this key. Therefore, even if the identity of a doctor is public, it cannot be used to commit fraud, as the communication protocols are based on the secret key.

3. Protocols

We consider four practical scenarios, for a patient to establish a secure communication channel with its doctor who is located at the same or a different location. They communicate by an established wireless or wired communication system. The protocols for other scenarios can be easily obtained with the protocols for these four typical scenarios.

3.1. Scenario 1

We consider a scenario (Fig. 2) where a patient who is situated in its own local place needs to consult a doctor due to its illness. In this scenario, both the patient and the doctor reside in the same communication domain, which is the home for both of them and is managed by HS.

1. Obtaining a ticket.
   The objective of this phase is to establish a secure communication channel between a patient and a doctor. This process requires mutual authentication between the patient and the doctor. This is done with the help of the HS who is trusted by the patient and the doctor. The output from this phase is a session key shared by the patient and the doctor. The following protocol is conducted by patient P and home-sever HS in order to obtain a service from doctor D. The patient P has a subliminal identity: P_j^s. For simplicity, we have omitted the subscript i.
   (a) $\mathsf{P} \rightarrow \mathsf{HS}$: $\mathsf{P}_1^s,\mathsf{HS},\mathsf{D},n_p,\mathsf{MAC}_{K_{p,hs}}(\mathsf{P}_1^s,\mathsf{HS},\mathsf{D},n_p)$
   (b) $\mathsf{HS} \rightarrow \mathsf{P}$: $\mathsf{HS},\mathsf{P}_1^s,n_p,[k_s,\mathsf{P}_2^s]_{K_{p,hs}},\mathsf{Ticket},\mathsf{MAC}_{K_{p,hs}}(\mathsf{HS},\mathsf{P}_1^s,n_p,k_s,\mathsf{P}_2^s,\mathsf{Ticket})$
   By $\mathsf{Ticket}_{p,d}$, we denote the ticket for the patient to communicate with the doctor, where

$$\mathsf{Ticket} = [k_s,\mathsf{P},\mathsf{P}_1^s,\mathsf{D},T]_{K_{d,hs}}$$

With the $\mathsf{Ticket}_{p,d}$, the patient P can contact the doctor for consultation.

2. Remote consultation phase.
   In this phase, the patient P communicates with the doctor D by sending $\mathsf{Ticket}_{p,d}$ to D. In turn, the doctor D replies to the patient P to confirm the establishment of the communication channel. The communication before P and D is conducted via the home server HS. Therefore, the subliminal identity of P can still be used to protect the patient. For simplicity, we omit the intermedia steps.
   (a) $\mathsf{P} \rightarrow (\mathsf{HS}) \rightarrow \mathsf{D}$: $\mathsf{P}_1^s,\mathsf{D},n_p,\mathsf{Ticket},\mathsf{MAC}_{k_s}(\mathsf{P}_1^s,\mathsf{D},n_p,\mathsf{Ticket})$
   (b) $\mathsf{D} \rightarrow (\mathsf{HS}) \rightarrow \mathsf{P}$: $\mathsf{D},\mathsf{P}_1^s,n_p,\mathsf{ok},\mathsf{MAC}_{k_s}(\mathsf{D},\mathsf{P}_1^s,n_p,\mathsf{ok})$
   (c) $\mathsf{P} \rightarrow (\mathsf{HS}) \rightarrow \mathsf{D}$: ⋯ Consultation request
   (d) $\mathsf{D} \rightarrow (\mathsf{HS}) \rightarrow \mathsf{P}$: ⋯ Reply
   This communication channel can last while there is a necessity. All the communication messages are encrypted with session key k_s, which provides the secure communication channel.

3.2. Scenario 2

We consider a scenario (Fig. 3) in which one of the patients (e.g. P) has traveled to another domain, rather than its own local area network, and requests a communication channel for the remote consultation session with the doctor who is still located at its home, which is managed by HS. Differing from the scenario 1, the patient P is unable to contact its home server HS directly. The communication with HS must be mediated by the visiting remote server RS, which does not share any secret key with P. Since HS and RS share a long-term secret key, the authentication communication flows are passed by RS to the home-server HS who then authenticates P in order to establish a consultation session with the doctor D.

1. Obtaining a ticket.
   In order to establish a secure communication channel between patient and doctor (similar to scenario 1), the patient needs to contact their home server to obtain a service ticket. In this scenario, the patient is located at the remote domain (RS) and needs to contact the remote server RS that acts as a mediator. The mutual authentication between P and D is done with the help of RS and HS, where the service ticket plays an important role. The main hurdle for this protocol to work is how to let the RS check whether P is a legitimate user. Since the RS does not hold any information about P, then it is unable to verify the authenticity of P, with the identification information provided by P. We solve this problem by allowing the P to encrypt the data with a temporary session key. Later, the HS can help provide such a key to the RS, therefore, the RS can check the authenticity of the P.
   (a) $\mathsf{P} \rightarrow \mathsf{RS}$: $\mathsf{P}_1^s,\mathsf{RS},\mathsf{D},\mathsf{HS},n_p,\mathsf{Token},\mathsf{MAC}_{tk}(\mathsf{P}_1^s,\mathsf{RS},\mathsf{D},\mathsf{HS},n_p,\mathsf{Token})$
   The patient P contacts the remote server RS as the first step to obtaining a service ticket. Here, $\mathsf{Token} = [\mathsf{P}_1^s,\mathsf{HS},\mathsf{RS},n_p]_{K_{p,hs}}$ and tk is a temporary key for the RS to verify the MAC after it has obtained the key from the HS. tk is derived from

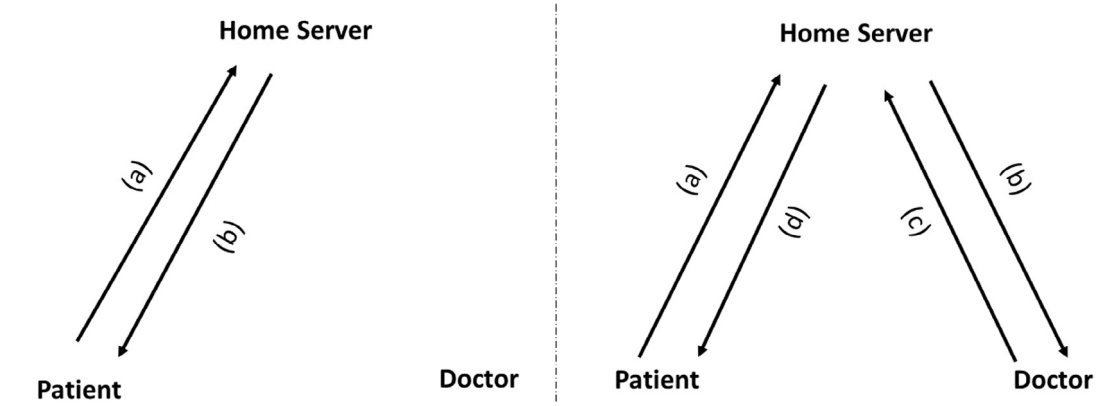

Fig. 2. Scenario 1. The figure on the left-hand side is for the patient ticket granting. The figure on the right-hand side is for the consultation session.

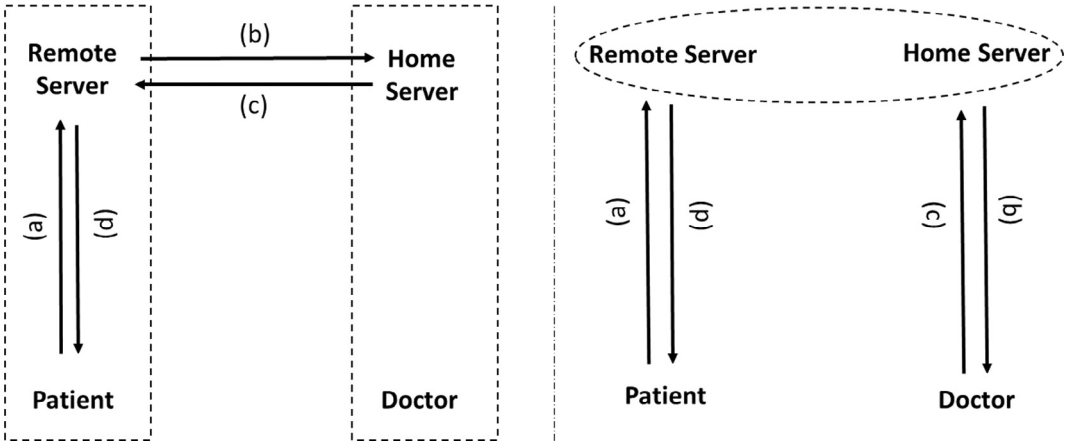

Fig. 3. Scenario 2. The figure on the left-hand side shows the case when the patient has traveled to the remote domain, while the doctor is still in its home domain. The figure on the right-hand side shows the consultation phase, which is done with the assistance of HS and RS.

$tk = f(K_{p,hs}, n_p, \mathsf{P})$, where $f(\cdot)$ is a cryptographic hash function.

(b) $\mathsf{RS} \rightarrow \mathsf{HS}$: $\mathsf{RS},\mathsf{HS},\mathsf{P}_1^s,\mathsf{D},n_p,n_{rs},\mathsf{Token},\mathsf{MAC}_{K_{rs,hs}}(\mathsf{RS},\mathsf{HS},\mathsf{P}_1^s,\mathsf{D},n_p,n_{rs},\mathsf{Token})$
In this step, RS forwards the Token to HS in order to get the temporary key *tk*. HS can decrypt Token and check the authenticity of P, and then compute the temporary key $tk = f(K_{p,hs}, n_p, \mathsf{P})$.

(c) $\mathsf{HS} \rightarrow \mathsf{RS}$: $\mathsf{HS},\mathsf{RS},\mathsf{P}_1^s,[\mathsf{P}_1^s,tk,n_p,n_{rs}]_{K_{rs,hs}},\mathsf{MAC}_{K_{rs,hs}}(\mathsf{HS},\mathsf{RS},\mathsf{P}_1^s,tk,n_p,n_{rs}),\mathsf{Package}$
where

$$\mathsf{Package} = [\mathsf{P}_2^s,\mathsf{D},k_s,\mathsf{Ticket}]_{K_{p,hs}},$$

$$\mathsf{Ticket} = [\mathsf{P},\mathsf{P}_1^s,\mathsf{D},k_s,T]_{K_{d,hs}}$$

Two main tasks in this step are: (1) HS needs to forward *tk* to RS. This was done by encrypting it along with the subliminal ID of P and nonces. n_{rs} was initialized by RS, hence it must be returned to RS to indicate the completion of the session between RS and HS. n_p was initialised by P, so it must be forwarded to RS so that it can be returned to P in the next step. (2) Package is sent to RS, so that it can forward it to P in the next step. The Package contains Ticket which is sent to P in the next step. In order for P and D to establish a secure channel for the consultation, a session key k_s is embedded in Package, to ensure both P and D will obtain it. The doctor D will obtain k_s from the service ticket Ticket when the P requests a consultation session to D.Note: The temporary key used in Step (a) plays a main role to achieve the full authentication of the protocol. Notice that to compute the temporary key, the patient needs to use his secret key (shared with his home server HS). However, the remote server RS cannot check its legitimacy in Step (a) yet, but just simply uses it. The authenticity of the temporary key is confirmed in Step (c) by the home server HS and the HS in turn informs the RS of the authenticity of the temporary key; therefore, the prior authentication phase in Step (a) is then confirmed. However, if failed to confirm the authenticity of the temporary key in Step (c), The RS will terminate the communication and output "Fail".

(d) $\mathsf{RS} \rightarrow \mathsf{P}$: $\mathsf{RS},\mathsf{P}_1^s,n_p,\mathsf{MAC}_{tk}(\mathsf{RS},\mathsf{P}_1^s,n_p),\mathsf{Package}$
In this step, Package is delivered to P by RS. P therefore obtains the service ticket Ticket and the secret session key k_s; hence P is ready to contact his doctor D for consultation. As a necessary matter, P needs to check if n_p is the same as the one initialized by itself in the first step. If the checking returns true, the entire protocol run is complete.

2. Remote consultation phase.
With the service ticket Ticket, P can then contact its doctor D. The protocol is the same as that in the first scenario. The only difference is that the communication between P and D is conducted with the aid from both RS and HS.

3.3. Scenario 3

In this case (Fig. 4), both patient P and doctor D have left their home domain and are situated in a distant location. The phase of ticket granting phase for the patient P is the same as that in the second scenario. However, since the doctor D has also left its home domain and is located at the remote domain managed by RS, the D needs to be authenticated by the HS. Therefore, in this case, we only present this part of the protocol.

In order to establish a secure communication channel between patient and doctor, the patient and the doctor need to contact their home server HS, since the remote server RS does not store any information about them. The mutual authentication among P and HS is done with the help of RS and HS, as the protocol given in Scenario 2. Accordingly, D needs to be authenticated by RS in order to establish a communication channel with its patients. Since RS does not hold any information about D, then it can verify the legitimacy of D only with the help of HS who registered D. The authentication protocol for D is similar to the patient authentication, while D does not require a subliminal ID and a service ticket.

1. Authentication phase.
The doctor authentication protocol is given as follows:

(a) $\mathsf{D} \rightarrow \mathsf{RS}$: $\mathsf{D},\mathsf{RS},\mathsf{HS},n_d,\mathsf{Token},\mathsf{MAC}_{tk}(\mathsf{D},\mathsf{RS},\mathsf{HS},n_d,\mathsf{Token})$
The doctor D contacts the remote server RS as the first step in order to be authenticated. Here, $\mathsf{Token} = [\mathsf{D},\mathsf{HS},\mathsf{RS},n_d]_{K_{d,hs}}$ and *tk* is a temporary key for the RS to verify the MAC after it has obtained the key from the HS. *tk* is derived from $tk = f(K_{d,hs}, n_d, \mathsf{D})$, where $f(\cdot)$ is a cryptographic hash function.

(b) $\mathsf{RS} \rightarrow \mathsf{HS}$: $\mathsf{RS},\mathsf{HS},\mathsf{D},n_d,n_{rs},\mathsf{Token},\mathsf{MAC}_{K_{rs,hs}}(\mathsf{RS},\mathsf{HS},\mathsf{D},n_d,n_{rs},\mathsf{Token})$
In this step, RS forwards the Token to HS in order to get the temporary key *tk*. The HS also checks the D′*s* information stored in HS in order to authenticate D.

(c) $\mathsf{HS} \rightarrow \mathsf{RS}$: $\mathsf{HS},\mathsf{RS},[\mathsf{D},tk,n_d,n_{rs}]_{K_{rs,hs}},\mathsf{MAC}_{K_{rs,hs}}(\mathsf{HS},\mathsf{RS},\mathsf{D},tk,n_d,n_{rs})$
This step captures the following task. HS forwards *tk* to RS in order to verify the authenticity of D by RS. This was done by encrypting it along with the ID of D and nonces. RS can now verify the MAC provided by D in the first step. This step also confirms that D is a legitimate user of HS.

(d) $\mathsf{RS} \rightarrow \mathsf{D}$: $\mathsf{D},\mathsf{RS},n_d,\mathsf{accept/reject},\mathsf{MAC}_{tk}(\mathsf{RS},\mathsf{D},n_d,\mathsf{accept/reject})$
The protocol ends after RS confirms "accept" or "reject" to D. In a case of "accept", RS will keep the D′*s* credential for future

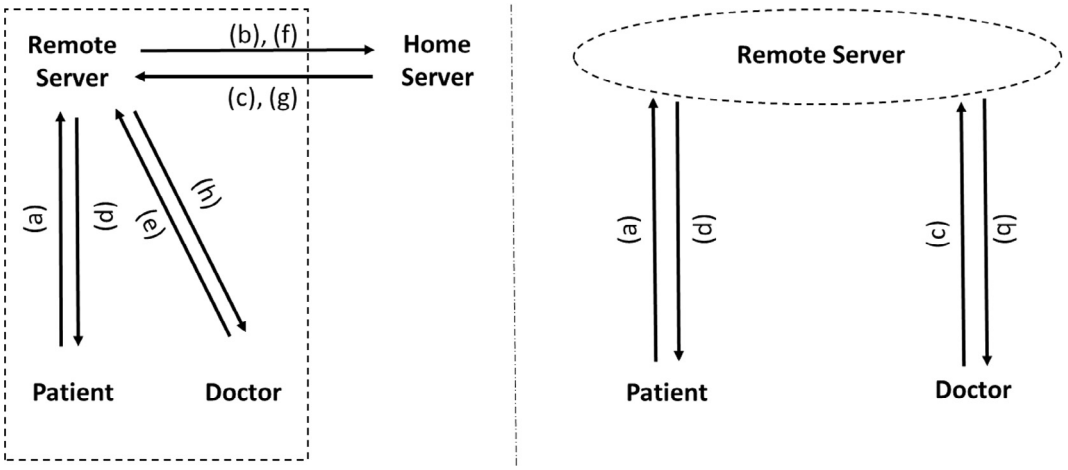

Fig. 4. Scenario 3. The figure on the left-hand side shows that the patient and the doctor both have left HS and are now located at the remote domain managed by RS. The figure on the right-hand side shows the consultation communication flows.

communication.

This protocol is essential for P to be able to access the service provided by D, since the future service to P requires RS to act as the hub to bridge P and D.

2. Remote consultation phase.

In this phase, the patient P communicates with the doctor D by sending $\mathsf{Ticket}_{p,d}$ to D. In turn, the doctor D replies to the patient P to confirm the establishment of the communication channel. The communication before P and D is conducted via the home server HS and the remote server RS. Notice that we have omitted the ticket granting phase for P as it is same as that in Scenario 2. Therefore, it is similar to Scenario 2, the subliminal identity of P can still be used to protect the patient. This requires a routing table which contains the information of P and D including their IDs, patient's subliminal ID, home domain and current location. We describe it later in this paper.

(a) $\mathsf{P} \rightarrow (\mathsf{RS}) \rightarrow \mathsf{D}$: $\mathsf{P}_1^s,\mathsf{D},n_p,\mathsf{Ticket},\mathsf{MAC}_{k_s}(\mathsf{P}_1^s,\mathsf{D},n_p,\mathsf{Ticket})$

(b) $\mathsf{D} \rightarrow (\mathsf{RS}) \rightarrow \mathsf{P}$: $\mathsf{D},\mathsf{P}_1^s,n_p,\mathsf{ok},\mathsf{MAC}_{k_s}(\mathsf{D},\mathsf{P}_1^s,n_p,\mathsf{ok})$

(c) $\mathsf{P} \rightarrow (\mathsf{RS}) \rightarrow \mathsf{D}$: ⋯ Consultation request

(d) $\mathsf{D} \rightarrow (\mathsf{RS}) \rightarrow \mathsf{P}$: ⋯ Reply

This communication channel can last while there is a necessity. All the communication messages are encrypted with session key k_s, which provides the secure communication channel.

3.4. Scenario 4

In this scenario (Fig. 5), suppose a patient P lives in a remote rural area and has registered with her home domain server RS. A doctor *D* resides in a major city and has also registered with his home domain server HS. P needs to travel to different places temporarily in which her communication requests are being forwarded to RS.

P, when is in her remote area, sends requests to her home server RS in order to establish remote consultation with D. The RS in turn contacts HS in order to obtain a service ticket for P. A secure communication session can be established through home servers RS and HS. The P's home server RS is trusted by D's home server HS in which her requests are being sent to. Since RS can authenticate P, also HS and RS are trusted each other, the authentication of P by HS is based on a mutual trust.

Once the authentication is done, the HS will issue a service ticket and send it to RS who can then forward it to P. Since D is located at his home domain and already registered, he does not need to be authenticated again.

1. Obtaining a ticket.

In this phase, the patient P needs to contact its home server RS to obtain a service ticket for a secure communication channel with the doctor who resides in HS. The RS will contact the HS in order to

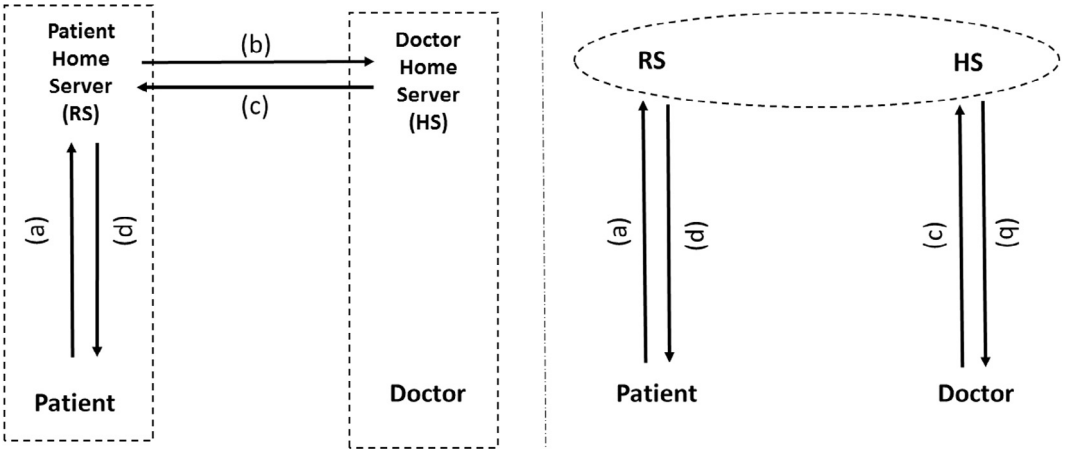

Fig. 5. Scenario 4. The figure on the left-hand side shows the case while both P and D have a separate home server and both reside in their own home. The figure on the right-hand side shows the consultation phase.

obtain a service ticket for P.

(a) $P \rightarrow RS: P_1^s,RS,D,HS,n_p,MAC_{p,rs}(P_1^s,RS,D,HS,n_p)$
The patient P contacts its home server RS as the first step to provide the information of the doctor's ID and its home domain, along with a nonce n_p. The RS authenticates P by verifying the MAC.

(b) $RS \rightarrow HS: RS,HS,[P,P_1^s,D]_{K_{hs,rs}},n_p,n_{rs},MAC_{K_{hs,rs}}(RS,HS,p,P_1^s,D,n_p,n_{hs})$
In this step, the patient's home server RS contacts the home server of D to request a service ticket for P to communicate with D.

(c) $HS \rightarrow RS: HS,RS,Package,MAC_{K_{hs,rs}}(HS,RS,Package)$ where

$$Package = [P_1^s,D,k_s,Ticket]_{K_{hs,rs}}$$

$$Ticket = [P,P_1^s,D,k_s,T]_{K_{d,hs}}$$

Package is sent to RS, so that it can forward it to P in the next step. Here, T is a timestamp. The Package contains Ticket which is sent to P in the next step. In order for P and D to establish a secure channel for the consultation, a session key k_s is embedded in Package, to ensure both P and D will obtain it. The doctor D will obtain k_s from the service ticket Ticket.

(d) $RS \rightarrow P: RS,P_1^s,[P_2^s]_{K_{p,rs}},n_p,Package,MAC_{K_{p,rs}}(RS,P_1^s,P_2^s,n_p,Package)$
In this step, Package is delivered to P by RS. P therefore obtains the service ticket Ticket and the secret session key k_s; hence P is ready to contact his doctor D for consultation. A new subliminal ID P_2^s for patient P is delivered to P in this step. As a necessary matter, P needs to check if n_p is the same as the one initialized by itself in the first step. If the checking returns true, the entire protocol run is complete.

2. Remote consultation phase.
In this phase, the patient P communicates with the doctor D by sending $Ticket_{p,d}$ to D. In turn, the doctor D replies to the patient P to confirm the establishment of the communication channel. The communication before P and D is conducted via the home server HS and the remote server RS.

(a) $P \rightarrow (RS) \rightarrow (HS) \rightarrow D: P_1^s,D,n_p,Ticket,MAC_{k_s}(P_1^s,D,n_p,Ticket)$
(b) $D \rightarrow (HS) \rightarrow (RS) \rightarrow P: D,P_1^s,n_p,ok,MAC_{k_s}(D,P_1^s,n_p,ok)$
(c) $P \rightarrow (RS) \rightarrow (HS) \rightarrow D: \cdots$ Consultation request
(d) $D \rightarrow (HS) \rightarrow (RS) \rightarrow P: \cdots$ Reply
This communication channel can last while there is a necessity. All the communication messages are encrypted with session key k_s, which provides the secure communication channel.

4. Routing tables

The routing tables play an important role for the protocols. Both the HS and RS need to maintain a routing table, respectively. The table should contain the information of the real IDs of users for both patients and doctors, the up-to-date subliminal IDs for patients, home servers of users and the current server. This is updated once a user is registered, a change of subliminal ID, and change of user location to a different domain. In Table 3, we provide an example for some cases of our protocol.

Table 3
Example of routing table.

ID	Sub ID	Home server	Current server
P1	$P1_i^s$	RS	RS
P2	$P2_j^s$	RS	HS
D1		HS	HS
D2		HS	RS
⋮	⋮	⋮	⋮

5. Clinical examples

In this section, we present two clinical examples for our protocols. We describe how the service should be delivered and how patients and doctors should be registered and managed in our system to capture the features of user mobility and distributed service.

For a distributed telemedicine system, using multiple servers to manage the services is essential to make the system efficient and practical for telemedical services. Therefore, we set up multiple servers in terms of geographical locations. We use the term "telemedicine servers" to name such service. In our protocols, telemedicine servers is also authentication servers for our security services. Therefore, the telemedicine servers are located at various geographical locations and also play the role of authentication server. In order to use the service, all doctors and patients have to register with their local telemedicine server. There are the following entities for our examples.

- Telemedicine servers: The servers which manage doctors and patients to provide authentication service and secure channels for communication.
- Doctors: They are medical specialists who provide special service to patients. They usually reside in large medical centers such as hospitals or specialized clinics. A doctor has registered with a telemedicine server at his home location for example. He can however travel to a different location while can still provide his service to his patients.
- Patients: Patients are these people who need to consult a doctor due to their illness. Patients have registered with a telemedicine server which is usually located at theirs home location. Patients might be located in rural and deprived areas and cannot access specialists easily. Any patient can travel to a different location and still obtain the requested consultation with a doctor.

Our protocols presented in this work capture the aforementioned application scenarios. The examples provided here elaborate two medical cases in order to sketch a comprehensive picture of our solutions' applicability.

Our protocols allow flexible communication networks such as 4G LTE mobile service, wired internet service, and wifi networks. Without loss of generality, we assume that the telemedicine servers are web-based servers, where the users have registered with their valid ID.

Example 1. In this example, we show that how a patient who lives in a rural area does not have to travel to a metropolitan city or a facilitated hospital in order to receive a medical consultation with a specialist. Technically, this example can be explained by Scenario 4, which has been presented earlier. Suppose that a doctor Donald who has registered with the telemedicine server - New York City Telemedicine Service, and a patient Petty who has registered with another telemedicine server - Kalawao County Telemedicine Service. Pretty can visit a local clinic for her ordinary check-ups or consultations with a general practitioner (GP); however, in chronic conditions or operations, her local GP cannot provide the professional advice. Then, GP refers Petty to visit the specialist, Donald. Therefore, she can request secure remote consultation to Donald, who is located in New York City.

With our protocol presented in Scenario 4, we can allow Petty and Donald to establish a secure and authenticated channel for an online consultation meeting. Petty only needs to contact the Kalawao County Telemedicine Service by the provided web service via a wired or wireless communication network. With the aid of the New York City Telemedicine Service, the Kalawao County Telemedicine Service can obtain a service ticket for Petty to consult Doctor Donald in a secure and authenticated channel.

Example 2. Now, we assume that Doctor Donald is currently treating a patient Penny due to her chronic medical condition. Suppose both

Penny and Donald have registered with the New York City Telemedicine Service; therefore, it is convenient for Penny to visit Donald in order to obtain his advice or treatment. However, as part of Penny's job, she has to travel to a different city Los Angeles. Our Scenario 2 captures this case perfectly. The procedure is as follows.

Penny needs to contact the Los Angeles Telemedicine Service through its web service. According to our Scenario 2, even Penny has not registered with the Los Angeles Telemedicine Service, she can successfully receive a service ticket which allows her to establish a secure and authenticated remote communication channel with Donald. Therefore, Penny can still consult her Doctor Donald while she resides outside her registered region.

By employing our solutions, we ensure that a patient who is in a remote area can still benefit from secure remote consultation meetings with a specialist via remote servers which can significantly reduce the time and cost involved in travelling long distances for medical appointments. We might highlight the advantage of distributed feature of our solutions. Any server in the system can serve as the point of attachment for patients and doctors; therefore, it solves the bottle-neck problem in the centralized schemes.

6. Security analysis and discussion

6.1. Security analysis

Under the aforementioned security assumptions, our protocols possess the following security properties we have defined earlier where we consider outside adversaries only, including the outsiders who are not registered with the system and the insiders who have been registered with their home server but are involved in the protocol execution. All adversaries can launch active attacks. For simplicity, we denote the adversary by $\mathscr{A}$.

Property 1. *The proposed protocol ensures the data confidentiality against the adversary* $\mathscr{A}$.

A legitimate user shares a long-term symmetric key with its home-server. Since the data transmitted between a user and its home server is encrypted with the long-term key in the ticket granting phase, to access the data, one must have the key. The adversary $\mathscr{A}$ can be a registered user (patient or doctor) and holds its own shared key with its home server, but it does not have others' keys. Therefore, it is impossible for $\mathscr{A}$ to decrypt data if the encryption scheme is secure.

For each session, there is a new session key issued for patient-doctor communication and all data flows are encrypted with this key; therefore, it is secure against $\mathscr{A}$. The confidentiality while a patient is in a remote area is achieved with the data encrypted under a temporary key issued by the patient. Hence, $\mathscr{A}$ does not hold this key.

The security is also due to the assumption that the home server is trusted to all its users (patients and doctors). This ensures that the data received by the home server will not be revealed to $\mathscr{A}$. Also, the final session key k_s which is selected by the home server and transmitted with the help of the home server (or remote server) cannot be obtained by the $\mathscr{A}$, as we have assumed that all underlying encryption algorithms are secure.

Property 2. *The proposed protocol ensures the data integrity against the adversary* $\mathscr{A}$.

The data integrity is obtained with the Message Authentication Code (MAC) which is based on a secure cryptographic hash function, such as SHA-1. All transmitted data are embedded in MAC with a MAC key, which is, by default, the long-term key shared by a user and its home server. This ensures that only those parties who have that key can compute and verify a MAC value. If MAC is secure, therefore, the integrity of the data transmission in our system is ensured.

Property 3. *The proposed protocols ensure the patient anonymity against the adversary* $\mathscr{A}$.

The user anonymity is achieved with the subliminal ID of the user. In our protocols, even the $\mathscr{A}$ who has its own key cannot know other users' identities, though they can obtain the identity of their communication partner. The reason is threefold: (1) A subliminal identity is used only for one round of the protocol and a new subliminal ID is transmitted to the user securely. (2) The new subliminal ID is encrypted by the home server, therefore, only the corresponding user can decrypt it. (3) Our protocols also offer patient untraceability, which means that a patient cannot be traced back with its previous communication transcripts. The reason is that the subliminal ID is updated when a communication session is completed. The new session uses a new subliminal ID.

Property 4. *The proposed protocols provide the freshness in communication against any replay attack.*

We utilize nonces in our protocols. The following facts support our claim. The communication session between two parties is accompanied with a new and random nonce and the nonce must be returned to the initiator of the nonce to ensure the completion of the session and ensure that the freshness can be verified. Since the different communication session has a different nonce, the information from the previous session cannot be applied to the current session as it has a different nonce. In addition, the adversary cannot use an old communication nonce in the future protocols; because the server keeps a table of all the nonces for a period of time. The freshness can be further enhanced by adding a timestamp T. Since it is a straightforward process, we omit it in this paper.

Property 5. *The proposed protocols provide the authentication service.*

Since MACs are used in our protocols, our protocols provide symmetric-key based authentication service. That is, for two parties who share a symmetric key can authenticate each other due to the fact: only the party who holds the key can compute the MAC value, therefore, the corresponding receiver of MAC value can verify it by using the same key.

The patient is granted a ticket, which is encrypted under doctor's key shared with its home server. If the doctor can decrypt it successfully, the doctor can be ensured that the ticket was created by its home server, which implicitly authenticates its home server. Also, if the doctor can successfully respond to the patient upon receiving the ticket, the patient then knows that the doctor is legitimate. Since the ticket contains the patient's ID, the doctor will know that the patient is legitimate.

6.2. Discussion

All our protocols can be referred to be two main phases: authentication & key establishment and tele-consultation. In the latter phase, the patient and the doctor are connected and establishingengaging a communication session. As we have mentioned earlier, our systems can be built on top of the existing telecommunication; therefore the handover communication protocol should still work with some modifications. Notice that since the patient is using a subliminal ID, the new domain cannot recognise it. The solution is to allow the patient's mobile device to contact the new domain while the mobile device has reached the overlapping area where both signals from Domain A and Domain B are available. What the patient's mobile device will do is to initialize an authentication session with Domain B with our existing authentication protocol, while the tele-consultation process continues. If the authentication is successful, then the server of Domain B will allow the handover. Since the communication flows between the patient and the doctor are encrypted, the server of Domain B will not be obtained the real content of communication. Notice that we only need the

Table 4
Summary of implementation and performance evaluation for Scenario 1.

Protocol phase	Computational cost
Authentication & key establishment	One decryption + two MAC computations
Challenge-response in consultation	Two MAC computations
On-going consultation	One encryption/decryption + one MAC computation

authentication part of our protocol to handle it and the protocols should be slightly modified.

We now briefly analyse the computation and performance of our protocols. Our protocols are designed with the aim of low computational complexity. The computations of our protocols are based on a symmetric-key encryption scheme such as AES and a MAC algorithm such as HMAC. Therefore, they are implemented in the real-world applications. Assuming the computational complexity of symmetric-key encryption and decryption are the same, we take the Scenario 1 as an example. The patient only needs to compute (summarized in Table 4)

- in the authentication and key establishment phase, one decryption and two MAC computations;
- in the consultation phase,
  - for the challenge-response (step (a) and (b)), two MAC computations;
  - for on-going consultation, one encryption/decryption and one MAC computation for each round.

The computational complexity for a doctor is even lower, as the doctor is passive and only needs to wait for receiving a valid ticket, without being involved in the first phase.

We might consider a situation, when the service provided by a home server and a remote server is interrupted due to an unforeseen reason. This is a system implementation matter and is out of scope of our work. However, if it does happen, the system could have a buffer, which stores some prior communication information which can be used when the communication is recovered. Another workable solution is to re-execute our protocol and establish a new communication session. The latter might happen, when the time of interruption has been too long.

The number of users involved in the system depends on a variety of measures including the network speed and power, and can be decided by the system administrator.

7. Conclusion

In this paper, we proposed several security protocols for practical telemedicine applications. Our protocols offer security properties of data confidentiality, patient anonymity (untraceability), data integrity, data freshness, and mutual authentication. Our protocols are featured with the capability of handling user mobility along with patient anonymity, which is not achieved in the previous studies. We presented four practical telemedicine scenarios, in which we showed that our protocols have provided all properties we predefined in our work. Our protocols are efficient in terms of computation cost and speed, as they require the symmetric-key cryptographic schemes only.

Conflict of interest

None.

References

[1] F. Rezaeibagha, K.T. Win, W. Susilo, A systematic literature review on security and privacy of electronic health record systems: technical perspectives, Health Inform. Manage. J. 44 (3) (2015) 23.
[2] R. Amin, S.H. Islam, G.P. Biswas, M.K. Khan, N. Kumar, An efficient and practical smart card based anonymity preserving user authentication scheme for TMIS using elliptic curve cryptography, J. Med. Syst. 39 (11) (2015) 180.
[3] S. Kumari, M.K. Khanand, X. Li, F. Wu, Design of a user anonymous password authentication scheme without smart card, Int. J. Commun. Syst., doi:http://dx.doi.org/10.1002/dac.2853.
[4] Z.Y. Wu, Y. Lee, F. Lai, H. Lee, Y. Chung, A secure authentication scheme for telecare medicine information systems, J. Med. Syst. 36 (3) (2012) 1529–1535.
[5] D. He, J. Chen, R. Zhang, A more secure authentication scheme for telecare medicine information systems, J. Med. Syst. 36 (3) (2012) 1989–1995, http://dx.doi.org/10.1007/s10916-011-9658-5.
[6] J. Wei, X. Hu, W. Liu, An improved authentication scheme for telecare medicine information systems, J. Med. Syst. 36 (6) (2012) 3597–3604.
[7] Q. Jiang, J. Ma, Z. Ma, G. Li, A privacy enhanced authentication scheme for telecare medical information systems, J. Med. Syst. 37 (1) (2013) 9897.
[8] Q. Jiang, J. Ma, X. Lu, Y. Tian, Robust chaotic map-based authentication and key agreement scheme with strong anonymity for telecare medicine information systems, J. Med. Syst. 38 (2) (2014) 12.
[9] D. Mishra, J. Srinivas, S. Mukhopadhyay, A secure and efficient chaotic map-based authenticated key agreement scheme for telecare medicine information systems, J. Med. Syst. 38 (10) (2014) 120.
[10] H. Yang, H. Kim, K. Mtonga, An efficient privacy-preserving authentication scheme with adaptive key evolution in remote health monitoring system, Peer-to-Peer Network. Appl. 8 (6) (2015) 1059–1069.
[11] Y. Chen, R. Liau, L. Chang, Applications of multi-channel safety authentication protocols in wireless networks, J. Med. Syst. 40 (1) (2016) 26:1–26:15.
[12] Q. Jiang, X. Lian, C. Yang, J. Ma, Y. Tian, Y. Yang, A bilinear pairing based anonymous authentication scheme in wireless body area networks for mhealth, J. Med. Syst. 40 (11) (2016) 231:1–231:10.
[13] S.M.M. Rahman, M.M. Masud, M.A. Hossain, A. Alelaiwi, M.M. Hassan, A. Alamri, Privacy preserving secure data exchange in mobile p2p cloud healthcare environment, Peer-to-Peer Network. Appl. (2015) 1–16, http://dx.doi.org/10.1007/s12083-015-0334-2.
[14] S. Chatterjee, S. Roy, A.K. Das, S. Chattopadhyay, N. Kumar, G.R. Alavalapati, K. Park, Y. Park, On the design of fine grained access control with user authentication scheme for telecare medicine information systems, IEEE Access 5 (2017) 7012–7030.
[15] A.K. Sutrala, A.K. Das, V. Odelu, M. Wazid, S. Kumari, Secure anonymity-preserving password-based user authentication and session key agreement scheme for telecare medicine information systems, Comp. Meth. Prog. Biomed. 135 (2016) 167–185.
[16] A. Chaturvedi, D. Mishra, S. Jangirala, S. Mukhopadhyay, A privacy preserving biometric-based three-factor remote user authenticated key agreement scheme, J. Inf. Sec. Appl. 32 (2017) 15–26.
[17] F. Rezaeibagha, Y. Mu, Distributed clinical data sharing via dynamic access-control policy transformation, Int. J. Med. Inf. 89 (2016) 25–31.
[18] F. Rezaeibagha, Y. Mu, W. Susilo, K.T. Win, Multi-authority security framework for scalable EHR systems, Int. J. Med. Eng. Inf. 8 (4) (2016) 390–408.