IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received April 22, 2018, accepted May 21, 2018, date of current version July 6, 2018.

Digital Object Identifier 10.1109/ACCESS.2018.2841972

Anonymous Authentication Scheme for Smart Cloud Based Healthcare Applications

ABID MEHMOOD, IYNKARAN NATGUNANATHAN, (Member, IEEE), YONG XIANG, (Senior Member, IEEE), HOWARD POSTON, AND YUSHU ZHANG, (Member, IEEE)

School of Information Technology, Deakin University, Burwood, VIC 3125, Australia

Corresponding author: Yong Xiang (yxiang@deakin.edu.au)

ABSTRACT Many smart healthcare applications are adopting cloud to provide services to patients. However, the sensitive data can be disclosed to the authentication server/service provider. Therefore, security and privacy are crucial to its success and deployment at large scale. Patients don't want to disclose their identities to the cloud server. One way to protect their identities from cloud server is anonymous authentication. The authentication process normally involves disclosing users' private information, such as username and password to the authentication server. If the patient can be linked or tracked by the authentication server or malicious adversaries by their requests, their privacy can be breached. Most of the existing privacy preserving health care applications provides anonymity from the adversaries. However, very few of them provide anonymity from the authentication server. In this paper, we have proposed a system which provides complete privacy and anonymity to the users of health care applications from adversaries and the authentication server. In our proposed authentication scheme, we have utilized rotating group signature scheme based on Elliptic curve cryptography to provide anonymity to the patients. To add an extra layer of protection, we have used The Onion Router to provide privacy at the network layer. The performance of our scheme is evaluated by theoretical analysis which demonstrates that it resists various attacks and provides several attractive security features.

INDEX TERMS Anonymous authentication, rotating group signature, elliptic curve cryptography, smart health applications.

I. INTRODUCTION

Recent advances in biosensors, wireless network and embedded systems have assisted the rapid development of a wide range of wearable and implantable sensors in the human body. To collect crucial health data such as blood pressure level, and heart rate, many smart phone based health applications have been developed in the recent past [1], [2]. The data from the sensors is sent to the cloud server, where hospitals have hosted their services for data processing. The data is analyzed to improve the level of healthcare given to the patients. An example of smart cloud based health applications is shown in Fig. 1. Ideally, patients want hospitals to assist them with high efficiency without revealing patients' identities.

The increasing necessity for massive computation and excessive amounts of storage, is driving the healthcare industry to use cloud based servers, because of many advantages they are offering, such as cost saving and scalability. However, sharing data on un-trusted clouds can put patients privacy at risk. Cloud computing can also result in serious cloud specific privacy issues such as losing physical control over your data. Unless we provide complete security and privacy,[1] patients will always hesitate to transfer their private or sensitive data to the cloud. There are some challenges which we must overcome for building a trustworthy and secure data storage/processing system for healthcare applications on the cloud. These challenges are as follows [3]

- Losing physical control: Outsourcing the data to the clouds reduces the capital and operational expenditure. However, the risks associated with losing physical control over data must not be ignored. We need to build

[1]Security and privacy of the data are interrelated. Security means providing three services i.e., confidentiality, integrity and availability. Confidentiality or privacy means concealing information from unauthorized parties and is one part of security. In this paper, by security and privacy we mean protecting the sensitive information of the user and we will use the terms security and privacy interchangeably.

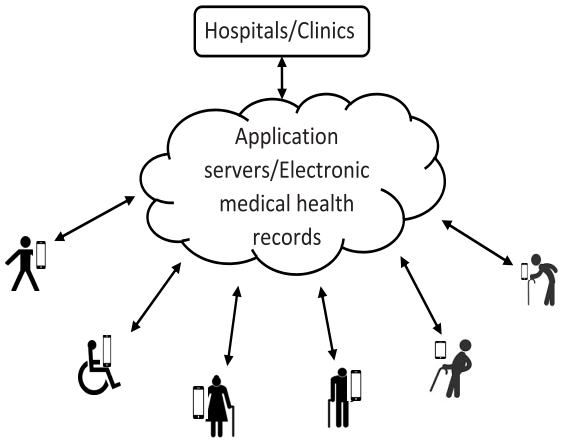

FIGURE 1. An example of typical smart healthcare system.

mechanisms to protect users' privacy and protect their confidentiality and integrity.

- Multi-tenancy: Due to virtualization, it is now possible that multiple customers share the same physical storage, with the help of some resource allocation policy. It is relatively easy for a malicious user to illegally access data that does not belong to him in such an environment.
- Privacy breach: Due to huge processing power, traditional mechanisms to protect an individual's privacy may not be sufficient. For instance, our online activities reveal a lot about us. Cloud server or any eavesdropper analysing our browsing habits and location footprints can be very dangerous.

In summary, due to the exponential use of smart devices large amounts of users' data are generated, stored, and processed on cloud increasing the risk of privacy violation. The smart devices and cloud servers not only collect personal information of the users such as a username, password, and telephone numbers but can also monitor users' activities such as shopping preferences and access history. All these risks associated with smart cloud based applications have led researchers to focus on applications with strong privacy [4]–[12].

A. MOTIVATION

To access the healthcare services online such as monitoring chronic diseases (i.e., Cancer, diabetes, asthma, dementia etc.), a patient must authenticate himself/herself. The authentication process normally involves revealing the identity of a user such as username and password. Based on the information provided to the cloud server during the authentication process, the patient can be linked or tracked using their access history or preferences. Recently, many privacy preserving authentication schemes have been proposed [4]–[9]. However, these schemes do not provide complete anonymity. The aim of our research is to allow patients to use healthcare services without revealing their identities or being tracked by an eavesdropper.

In many scenarios, users pay the membership fee for a certain amount of time to be eligible to use the services until their membership expires. For instance, a member of ambulance service can use ambulance services whenever required throughout his/her membership. Following are some examples where the proposed scheme can be used effectively

1) The patients can book an appointment with a healthcare professional or call an ambulance in case of emergency using the smart phone without revealing their identities.
2) In remote health care monitoring, information of interest like blood pressure level or heart rate is gathered by the sensors attached to the body and transmitted by a controller (mobile devices or personal digital assistants) to a server where it is processed. Consider an example of a patient where an application raises an alarm automatically when the readings from the sensors go above the threshold. For example, relevant authorities (i.e., ambulance service) can be notified when patient's blood pressure goes above the threshold with the chronic heart disease or whenever a patient with dementia in assisted living facilities goes out of their defined boundaries on a given map.

The identity privacy of a patient can be breached directly or indirectly. Direct privacy violation involves revealing personal information such as username, biometric features etc., used during the authentication process whereas, indirect violation involves extracting the information from the hidden patterns such as analysing the Internet traffic or patient's preferences.

To preserve identity privacy of the patient, many additional relationships between patient and authentication server must also be hidden during the communication rounds between them. We aim to achieve the following features in our proposed anonymous authentication scheme:

- Anonymity: The patient must be able to authenticate anonymously with authentication server without revealing his personal details.
- Unlinkability: The authentication server and eavesdropper should not be able to link whether two or more authentication requests are coming from the same patient. if multiple requests by the same patient can be linked together by an eavesdropper or authentication server, the patient may be identified.
- Traceability: It is important to identify the user in some cases. Providing complete anonymity to the patients sounds very appealing but also has some serious implications. Total anonymity for the system can be achieved at the cost of compromising auditing process which means auditing for the system is not possible. For example, a malicious patient can overload the system with fake appointment bookings or unnecessary ambulance calls and we will have no way to find the source of the malicious request. However, for practical application scenarios, this is not the ideal case. We need a mechanism which provides maximum anonymity to the patients as long as they do not misbehave. In case the

patient misbehaves, his identity should be revealed and his membership could be revoked if necessary.
- Resistance to attacks: It is important for a system to be resistant to all possible attacks i.e., eavesdropping, replay attacks, man in the middle attack etc.
- Integrity: The transmitting messages should be verifiable by the receiving party.

The primary goal of this paper is to provide identity privacy for smart cloud based healthcare applications by providing anonymous authentication. The proposed scheme can be generalized and applied to other cloud based applications. In some scenarios, user can potentially be identified by the operations performed on a specific set of data over time. Therefore, the proposed scheme is most suitable for scenarios where the specific user cannot be identified by operations over the data. It is worth noting that other issues such as location privacy and query privacy in smart health application are equally important. However, the study of those issues is outside the scope of this work.

The rest of the paper is organized as follows. Section II reviews the related work on anonymous authentication. Section III describes the background of bilinear pairings and group signatures. The system model of proposed scheme is presented in Section IV. The detailed construction of proposed scheme is described in Section V. We evaluate the security of proposed scheme in Section VI-A and complexity analysis in Section VI-B. Section VII concludes this work.

II. RELATED WORK

The related work on anonymous authentication schemes can be broadly classified into public key cryptosystems (*PKC*) based schemes [13]–[22], identity based cryptosystems schemes [23]–[28], pseudonyms based schemes [29], [30], combined scheme [31] using both identity based encryption and pseudonyms, and application oriented schemes [32]–[47].

Anonymous authentication schemes based on *PKC* in [13] and [14] were infeasible for mobile networks because of the computational resources required by *PKC* modular exponentiation, which consume more resources than what a mobile device can offer. To minimize the computational requirements, various anonymous authentication schemes based on elliptic curve cryptosystem (*ECC*) have been proposed [15]–[22], which have better performance because of the smaller key size used in *ECC*.

The performance of *ECC* based schemes are enhanced by identity based cryptosystems [23]–[28] over *ECC*. Unlike the traditional *PKC*, the identity based cryptosystems exploit public identity such as ID or email address as the user's public key to eliminate the cost related to the management of public key certificates, which is often desirable in mobile environments. However, the schemes built upon the identity based cryptosystems have two major drawbacks [28]. Firstly, the identity based authentication systems suffer from the key escrow problem, where a key needed to encrypt/decrypt is held in escrow so that under certain circumstances, an authorized party may gain access to the key. The key escrow systems are considered as a security risk as the escrow agent holding all the cryptographic keys can be a single failure point or may leak information. Secondly, once a user's private key is compromised, it is very hard to revoke the user [26], [27].

Anonymous authentication schemes are also emerging through the usage of pseudonyms and other techniques to protect privacy [29], [30]. Lin *et al.* [29] and Sun *et al.* [30] gave solutions to privacy and emergency response based on the anonymous credential, pseudo-random number generator, and the knowledge of proof for peer to peer systems. However, when directly applied to the distributed healthcare system, these schemes are impractical due to heavy computational overhead. In [31], Lin *et al.* proposed a strong privacy preserving scheme against global eavesdropping (*SAGE*), utilizing pseudonyms and identity based encryption to increase the level of security. This scheme provides both content and context oriented privacy resisting strong global adversary.

In recent years, two factor (password and smartcard) authentication schemes [32]–[35] have been proposed. However, these schemes have limitations as the smartcard and password both can be lost, stolen or duplicated. Also, some of these schemes [32], [33] require the server to maintain a password table for verification purposes, making them suffer from some possible attacks such password disclosure attacks and server spoofing attacks. To tackle the problems in two factor based authentication schemes, three factor (biometric, password and smart card) [36]–[39], authentication schemes have been proposed. To enhance the security, biometric characteristics are employed as a third factor to design a strong authentication scheme. These schemes are designed for smart cards. However, smart cards can't perform more sophisticated tasks like connecting to the server or communication with the server therefore, these schemes are not suitable for authentication using smart phones.

In [40], an anonymous authentication scheme in the cloud environment for e-health has been proposed. The scheme in [40] is based on blind signatures which allow users to consume cloud services anonymously. The scheme does not provide any details about user registration and revocation. The security analysis of the scheme is not discussed in detail.

In [41] and [42], anonymous authentication schemes for wireless body area networks (*WBANs*) were proposed. The scheme in [41] is based on bilinear pairings and it proposed an efficient anonymous authenticated key agreement scheme for *WBANs*. The scheme in [42] is based on certificateless encryption, which was designed to eliminate the drawbacks of the *PKI* based schemes. It does not require digital certificate and identity based encryption, i.e., no key escrow problem. The schemes in [41] and [42] were both efficient, however, revocation was not clearly defined.

In [43], an anonymous authentication scheme was proposed for wireless networks using Verifier-Local

revocation (*VLR*) group signature scheme. However, the scheme in [43] is vulnerable to replay attacks and a malicious Group Manager can impersonate a user.

In addition, the anonymous authentication schemes based on group signatures have been proposed in many areas [44]–[47]. The group signature schemes have been widely used in vehicular ad hoc networks (*VANETs*) to realize anonymous authentication [44]–[46]. To improve the contextual awareness, the vehicles communicate with each other by broadcasting safety messages with position beacons. And the group signatures are used to protect the identity of the users sending safety messages to protect the location and identity privacy. In [47], the group signatures are employed to preserve privacy in pervasive social networking (*PSN*), where *PSN* supports instant social activities at anytime and anywhere. The anonymous authentication is achieved by authenticating the trust levels rather than the identities of nodes. The scheme achieves anonymity and conditional traceability by utilizing group signatures and with the support of Trusted Authority (*TA*). However, the anonymous authentication schemes in [44]–[47] are very different in context to our proposed scheme.

Most of these aforementioned schemes provide anonymity from the eavesdropper. However, the service provider is still able to extract the real identity of the user to provide him access to the services and can link multiple messages coming from the same user.

A. OUR CONTRIBUTION

To provide the services to the patients while maintaining privacy, we propose an anonymous authentication scheme utilizing rotating group signatures based on *ECC*, which is both secure and efficient. Our contributions can be summarized as the follows:

- We have utilized a rotating group signature scheme based on *ECC* [48], to provide anonymous authentication to smart cloud based healthcare application users to prevent them from untrusted authentication server and against eavesdropping.
- Our scheme provides a mechanism for traceability with minimal compromise on privacy, in case of a malicious activity. Each group shares an expiration date and members renew their keys regularly, to minimize authentication time or the need to reveal a member's past authentications when their key is revoked.
- Anonymous authentication systems are usually considered in isolation. A service provider operating over the Internet can link subsequent requests by IP address, potentially connecting them to a physical location or an individual. We use *TOR* to provide anonymity on the network level, and minimize the information available to a service provider. A *TOR* hidden service cannot be accessed in a non-anonymous fashion, and complements anonymous authentication schemes nicely, leaving the service provider with little information to link subsequent connections.

III. PRELIMINARIES

In this section, we will review the mathematical background and security objectives, which are used as the building blocks for the proposed anonymous authentication scheme.

A. BILINEAR PAIRINGS

A pairing based cryptography is based on pairing between elements of two cryptographic groups to a third cryptographic group to construct or analyze a cryptographic system. Bilinear pairings are defined by the set of three abelian groups G_1, G_2 and G_T over a finite field $\mathbb{Z}_n$ together with a deterministic function e, called a bilinear map with following properties

1) G_1 and G_2 are two cyclic groups of prime order p.
2) P_1 is a generator of G_1 and P_2 is generator of G_2
3) e is a computable map $e : G_1 \times G_2 \rightarrow G_T$ with the following properties
   a) Bilinearity: for all $P \in G_1, Q \in G_2$ and $a, b \in \mathbb{Z}$, $e(P^a, Q^b) = e(P, Q)^{ab}$
   b) Non-degeneracy: $e(P_1, P_2) \neq 1$.

Bilinear pairing can use ranges of pairing groups but if these groups are elliptic curves, this mapping has a very useful interaction with the discrete logarithm problem. Elliptic curve cryptography was proposed in 1985 by Miller [49]. It has been used widely to provide security in the algorithms. ECC can be implemented in many different ways rather than as a single encryption algorithm. The security strength of ECC is much stronger than other public key encryptions schemes like RSA and Diffie-Hellman [17]. ECC provides stronger security with smaller key sizes. For instance, 160 bits in ECC provides the same security level as 1024 bits key in RSA. Due to the smaller key size and scalar multiplication used to provide security, ECC is much more efficient than RSA and Diffie-Hellman.

B. GROUP SIGNATURES

A group signature scheme is based on digital signatures [50]. Group signatures allow a group manager to issue keys to a number of members. A member can then use that key to prove membership in that group without identifying himself. Anyone with the public key may verify this proof. Additionally traceable group signatures allow the group manager to open a proof of membership to determine which member was responsible, and to revoke a member.

1) KEYS AND SIGNATURES

In any traceable group signature scheme there are a number of keys and transcripts involved

- G_M: The group manager key which is used to issue member keys and open proofs of the membership if needed to reveal which member signed the message
- G_P: The group public key which is known to everyone in the protocol and used in every step

- C_i: A transcript of key issue process for user i, used to identify and revoke members after opening a proof of membership
- M_i: A key, known only to one member of the group, generated by the issue process, and used to prove group membership
- P_j: A proof of membership during j^{th} authentication, generated during the prove process
- I_j: A piece of the transcript C_i extracted from the proof P_j
- T_i: A trace of a member, generated to revoke a member and used to verify a proof of membership comes from a member in good standing
- R: Random numbers, used in most phases of the protocol

2) PHASES

- Generate: The group manager generates a group manager key and derives a group public key from it.

$$R \rightarrow G_M$$
$$R, G_M \rightarrow G_P$$

- Issue: A member and a group manager together generate a key from the group manager key and public key, leaving the manager with a membership transcript and the member with a member key

$$R, G_M, G_P \rightarrow C_i, M_i$$

- Prove: A member demonstrates their possession of a member key to a third party, leaving both with the same proof of membership.

$$R, M_i \rightarrow P_j$$

- Verify: The third party may then check that a proof of membership against a group public key.

$$P_j, G_P$$

- Open: The group manager may extract identifying information from a proof of membership, revealing which member did the proving.

$$P_j, G_M \rightarrow I_j$$

- Revoke:The group manager extracts information from the issue transcript so that third parties may check proofs of membership for particular members

$$C_i, G_M \rightarrow T_i$$

- Trace: A third party, given the tracing information from above, may check a proof of membership for the member.

$$T_i, G_P \rightarrow P_j$$

IV. SYSTEM MODEL

In this paper, we have considered a system model for a hospital where much of hospital's infrastructure is hosted on the public cloud to reduce the cost. Hosting the services on the cloud also increases the risk of privacy breach. In order to protect the identity of the patient during the authentication process, we have proposed an anonymous authentication model for smart health applications which are used to access the services hosted on the public cloud. Table 1 summarizes the notations used hereafter in this paper to improve the readability.

TABLE 1. Notations used in this paper.

Notations	Descriptions
K_T	Long term keypair, the secret key K_{T_S} is known only to *GM* and the public key K_{T_P} is known to everyone in the protocol
G_M^n	The nth group signature master key, known to *GM* and *RS*
G_P^n	The nth group signature public key, known to everyone in the protocol
K_E^n	The nth log encryption key, the private key $K_{E_S}^N$ is known only to *GM*, the public key $K_{E_P}^N$ is known to *RS* and *CSP*
I^n	The issue date of a group, when it starts being valid
E^n	The expiry date of a group, when it stops being valid
M_i^n	A signing key for nth group held by member i
C_i^n	A transcript of the process that generated M_i^n that can be used along with G_M^n to produce a revocation for member i
c	A randomly generated challenge
Z	The request to be authenticated
$H(.)$	A secure one way hash function

The system model is described as follows

- *Trent* represents hospital/clinic, sometimes also referred as Group Manager (*GM*), who has outsourced much of its infrastructure to the cloud also referred as Cloud Service Provider (*CSP*).
- *Trent* has also hosted a Registration Server (*RS*), which provides registration services to *Trent*'s clients.
- *CSP* provides services to *Trent*'s clients, but must not be able to extract any personal information about *Trent*'s clients.
- *Alice*, *Bob*, *Carol*, and *Dave* are clients of *Trent*, utilizing *CSP*'s services.
- *Mallory* might control *Trent*'s, *CSP*'s and *RS*'s Internet connection, alternatively *Mallory* might control any of *Trent*'s clients' Internet connection.

From a particular security perspective, we have split up the roles to (*GM*) and registration server (*RS*). The *GM* is responsible for system initialization, key generation, and auditing whereas the *RS* is responsible only for patient registration. Using this approach, the *GM* only has to be on-line to generate keys and during the auditing process. The *RS* has to be on-line all the time but it does not need to have access to the private key of the *GM* which increases the security of the system. Fig. 2 shows the framework for the proposed anonymous authentication scheme. The identity of the patient is completely hidden from the *CSP* during the authentication process. Only in case of a malicious user or an emergency the identity of the patient can be revealed with the help of the *GM*.

A. THREAT MODEL

In this work, we consider the *CSP* as an un-trusted entity. This means that while the *CSP* follows the designed protocol, it may try to gain as much information about the user as

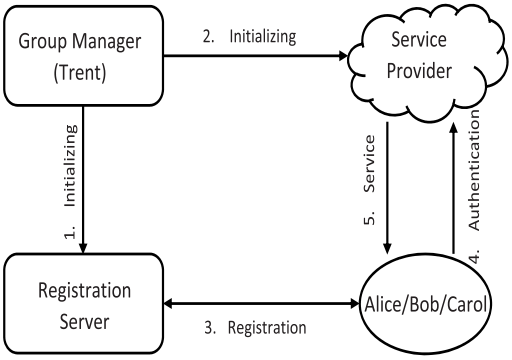

FIGURE 2. Framework for Anonymous authentication.

possible and keep a log of that. Specifically, two types of adversaries are considered.

- Weak Adversary: A weak adversary may be an outsider without access to the system. They are able to eavesdrop the communication channel but unable to alter any messages.
- Strong Adversary: A strong adversary may be a legitimate but malicious user with access to the system. They can eavesdrop the communication channel and may also be able to modify the messages.

The claims made by our protocol (i.e., anonymity, unlinkability, traceability) are resistant to attacks, and integrity will be tested against both types of adversaries. In Section VI, the proofs of security are provided, which demonstrate that our protocol meets/exceeds the security level of similar schemes.

B. PRIVACY GOALS

During the authentication process, the *CSP* can collect the user information by keeping the access logs including access locations, data access patterns and the services received. This record contains private information about the user which should be managed securely. The secure management of the users' private information is very important as the malicious administrator can pose a serious threat to users' privacy by leaking personal information. In this paper, our privacy goal is to keep service provider unaware from the private information of users such as access tokens and access history. Our privacy goals for the system are as follows:

- *Alice*[2] must be able to prove to *CSP* that she is an authorized client of *Trent*.
- *CSP* must not be able to determine the identity of *Alice*.
- Only *Trent* should be able link two performances of the protocol with the same client together.
- *Alice*'s account with *Trent* must be able to expire without compromising her privacy.
- *CSP* must keep a log such that only *Trent* can link multiple requests to each other if required.
- *CSP* must not be able to fake the log.
- *RS* must keep a log such that only *Trent* can link clients to requests.
- *Mallory* must not be able to pretend to be *RS*, *CSP*, *Trent*, or *Alice*.
- *Alice* must not be able to pretend to be *Bob* from *Trent*'s perspective.
- *Trent* should be able to revoke *Alice*'s authorization to use the service with minimal privacy compromise.
- *Alice* should be able to authenticate to *CSP* even when *RS* in unavailable.

[2]Here *Alice* represents one of the *Trent*'s clients.

C. PLATFORM CONSTRAINTS

- *Alice* has an off-the-shelf smartphone connected to the Internet.
- *RS* has a privately hosted server with maximum availability. We assume that the server hosted is very secure.
- *Trent* uses a dedicated computer that is only connected to a network during key generation and revocation.
- *CSP* uses the cheapest cloud service with maximum availability.

V. PROPOSED SCHEME

A. INITIALIZATION

The system initialization is done once, when *Trent* sets up the system.

1) *Trent* generates a long term private-public signing key pair, K_{Ts} and K_{Tp}.
2) *Trent* gives public half of that key i.e., K_{Tp} to *CSP* and *RS*.
3) The implementation allows the system to use range of pairings, the participants must first agree upon which pairings to use i.e., $(p, P_1, G_1, P_2, G_2, G_T, e)$.
4) *Trent* also selects a hash function $H : \{0,1\}^* \rightarrow \mathbb{Z}_p$ and shares it with the participants in the protocol.

B. KEY GENERATION

When setting up the system *Trent* generates a new group key that is valid for some time. The implementation uses group signature scheme based on bilinear pairings.

1) *Trent* generates the master group key G_M^n, and generates/derives from that the public group key G_P^n. The master group key is simply a random number γ:

$$G_M^n \leftarrow \gamma \leftarrow \mathbb{Z}_n \tag{1}$$

The group public key is partially random:

$$Q \leftarrow G_1 \tag{2}$$

and partially derived from the group manager key:

$$R \leftarrow (P_2)^{\gamma} \tag{3}$$

hence, the group public key is

$$G_P^n = (Q, R) \tag{4}$$

2) *Trent* make a signed certificate $S_T(G_P^n, I, E)$ with K_{Ts} containing,
   - G_P^n
   - Issue Date
   - An Expiry Date
3) Using the *CSP's* public key, *Trent* encrypts and sends that certificate to *CSP* and encrypts (using *RS's* public key) and sends the same certificate along with G_M^n to *RS*.
4) *CSP* and *RS* decrypt and verify that the certificate is actually from *Trent* with the help of K_{Tp} and *RS* verifies that G_M^n goes with G_P^n.

It is worth nothing that the same process for key generation is needed every time *Trent* has to update the group keys.

C. REGISTRATION

When *Alice* joins the group she registers with *RS* to get a key for the current group.

1) *Alice* connects to *RS* over a secure channel.
2) *Alice* demonstrates her identity to *RS* with, for instance, a username and password.
3) *RS* verifies that *Alice* is authorized to use *CSP*'s services until the expiry date of the current group.
4) *RS* shares with *Alice* the bilinear pairing parameters from the initialization phase which are $(P_1, G_1, P_2, G_2, G_T, e)$.
5) *RS* sends *Alice* the signed certificate $S_T(G_P^n, I^n, E^n,)$ from the key generation phase.
6) *Alice* generate a random number,

$$x \leftarrow \mathbb{Z}_n \tag{5}$$

then sends $(P_1)^x$ to the *RS* who generates another random number,

$$t \leftarrow \mathbb{Z}_n \tag{6}$$

RS derives variable A from it

$$A \leftarrow ((P_1)^x.Q)^{(\gamma+t)^{-1}} \tag{7}$$

and sends A and t back to *Alice*, who can verify that she has successfully joined by computing the following equation:

$$e((P_1)^x.Q, P_2) = e(A, R.(P_2)^t) \tag{8}$$

which can be proved by substituting the values of A and R on the R.H.S of Eq. (8) as follows:

$$\begin{aligned} e((P_1)^x.Q, P_2) &\stackrel{?}{=} e((P_1)^x.Q^{(\gamma+t)^{-1}}, P_2^{(\gamma+t)}) \\ e((P_1)^x.Q, P_2) &\stackrel{?}{=} e((P_1)^x.Q, P_2)^{((\gamma+t)^{-1})(\gamma+t)} \\ e((P_1)^x.Q, P_2) &\stackrel{?}{=} e((P_1)^x.Q, P_2) \end{aligned} \tag{9}$$

The registration process leaves *Alice* with a member key,

$$M_i^n \leftarrow (x, t, A) \tag{10}$$

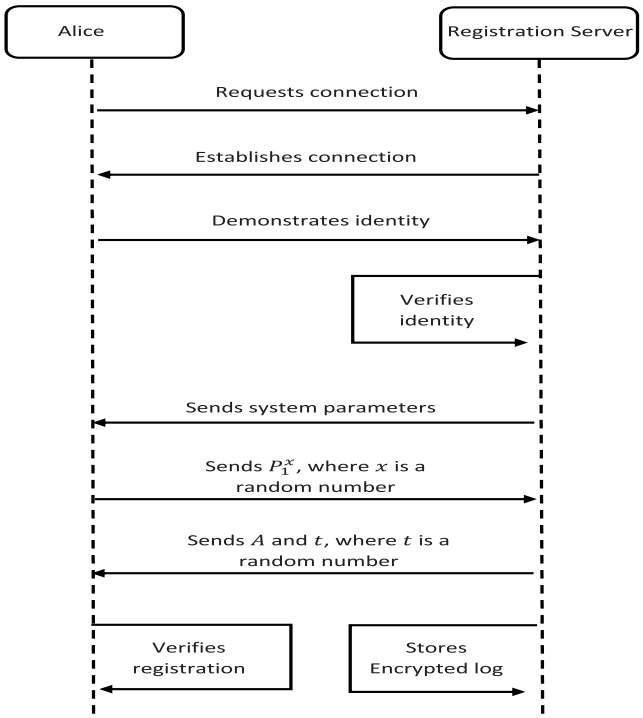

FIGURE 3. Member registration process.

and the *RS* with a transcript,

$$C_i^n \leftarrow ((P_1)^x, t, A) \tag{11}$$

7) *RS* encrypts the transcript C_i^n along with *Alice*'s identity using K_E and records it in the registration log.

Membership registration process is shown in Fig. 3. It is important to note that when a user *Carol* has to renew her keys, she has to go through the same process described in registration phase.

D. AUTHENTICATION

When *Alice* wishes to request a service represented by Z from *CSP*, she begins the authentication process:

1) *Alice* connects to *CSP* over an anonymous network.
2) *Alice* sends the group key to authenticate with.
3) *CSP* verifies the group key matches a stored certificate from *Trent* and that the certificate is not expired by checking the expiry E^n.
4) *CSP* generates a random number *RC* and sends it to *Alice*.

$$RC \leftarrow \mathbb{Z}_n \tag{12}$$

5) *Alice* and *CSP* perform a zero knowledge protocol to prove knowledge of x and/or A without revealing x or A as follows
   - *Alice* generates random r

$$r \leftarrow \mathbb{Z}_n \tag{13}$$

- *Alice* calculates a, b, c, and d as follows

$$a = Q.r \tag{14}$$
$$b = (P_1)^x.Q.r \tag{15}$$
$$d = r.A \tag{16}$$
$$c = H(e(d, RC.R.(P_2)^t), Z) \tag{17}$$

- *Alice* generates signature σ as follows

$$\sigma = (a, b, c, d) \tag{18}$$

- *Alice* sends (σ, Z) to *CSP*.
- *CSP* calculates

$$c \stackrel{?}{=} H(e(b, RC.P_2), Z) \tag{19}$$

and continues to next step if the equality holds true. *CSP* terminates the connection with *Alice* otherwise. It is straightforward to check that the verification equality holds for a valid signature. We can also prove that our proposed scheme satisfies completeness:

$$\begin{aligned}
&e(d, RC.R.(P_2)^t) \stackrel{?}{=} e(b, RC.P_2)\\
&e(d, RC.R.(P_2)^t) \stackrel{?}{=} e((P_1)^x.Q.r, RC.P_2)\\
&e(d, RC.R.(P_2)^t) \stackrel{?}{=} e((P_1)^x.Q, P_2)^{RC.r}\\
&e(d, RC.R.(P_2)^t) \stackrel{?}{=} e(A, R.(P_2)^t)^{RC.r}\\
&e(d, RC.R.(P_2)^t) \stackrel{?}{=} e(r.A, RC.R.(P_2)^t)\\
&e(d, RC.R.(P_2)^t) \stackrel{?}{=} e(d, RC.R.(P_2)^t)
\end{aligned} \tag{20}$$

- *CSP* checks that *Alice* didn't perform the protocol with the revoked key. The details of the revocation will be discussed in the Revocation section.

6) *CSP* performs the requested service represented by Z for *Alice*.
7) *CSP* encrypts (a, d, Z) with K_E and store it in the audit log so that only *Trent* may read it.

The authentication process for a member is shown in Fig. 4.

E. REVOCATION

When group manager *Trent* discovers that a user, let's say *Dave* has been abusing the service, he revokes *Dave*'s key. It is important to note, key revocation necessarily allows an unscrupulous *CSP* who keeps an unencrypted log to link all of *Dave*'s connections since the last key generation, so it should be used sparingly and not as the primary means of terminating a client's service. Membership revocation process is shown in Fig. 5 and described as follows

1) *Trent* requests the registration log from *RS* and decrypts it.
2) *Trent* requests the audit log from *CSP* and decrypts it to get (a, d, Z)
3) *Trent* uses G_M^n and the list of C_i^n to extract which M_i^n signed the message and consumed which service.
4) For each record in registration log, *Trent* tests $d.Q \stackrel{?}{=} a.A$
5) If a match is found, *Trent* computes the corresponding $(P_1)^x$ and adds it to the revocation log
6) To revoke *Dave*'s membership key, *Trent* shares the

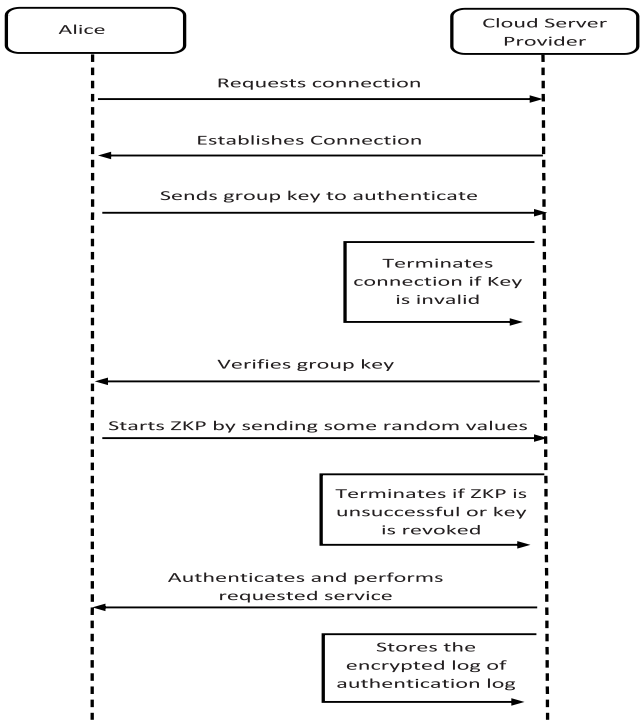

FIGURE 4. Member authentication process.

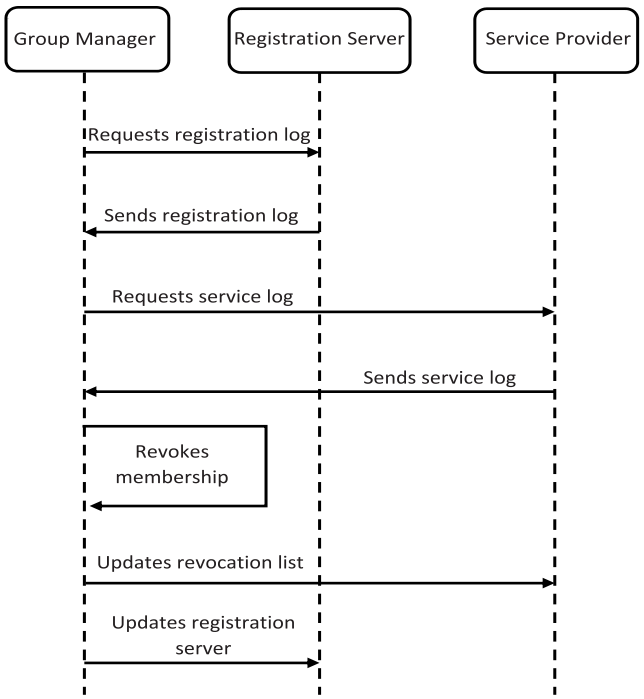

FIGURE 5. Membership revocation procedure.

tracing information for *Dave*, which is $T_i \leftarrow (P_1)^x$, with *CSP* in order to allow *CSP* to detect proofs of membership created by *Dave* and refuse providing services to him.

7) *Trent* sends *Dave*'s identity over a secure channel to *RS*, who removes *Dave* from the list of acceptable clients.

F. NETWORK LAYER ANONYMITY WITH TOR

The Onion Router (*TOR*) project is widely used to provide privacy at the network layer [51]. Internet data packets have two parts: a data payload (actual message i.e., email, audio file etc.) and a header used for routing. Even if we encrypt the data payload, data header can reveal some information about source, destination, time etc. Instead of using a direct connection, Internet users employ TOR network by connecting through a series of virtual tunnels. TOR protects its users against traffic analysis attacks. Traffic analysis can be used to infer who is talking to whom over a public network.

TOR protects privacy by building a private network pathway. The user's software incrementally builds a circuit of encrypted connections through relays on the network. The circuit is extended one hop at a time, and each relay along the way knows which relay the data is coming from and to which relay the data is going to. No individual relay is able to know the complete path the data packet has taken.

TOR bounces connections through 3 relay nodes. They are called entry guards, middle relay nodes, and exit nodes. Each of these nodes has a specific role to play. The entry points to the *TOR* network are called entry guards. Middle relay nodes are used to transport traffic from the entry guards to exit nodes. It also prevents the entry guard node and exit node from knowing each other. The exit point of the *TOR* network is called an exit node. These exit nodes send traffic to the final destination intended by the client. Each node in the *TOR* network knows the key for decryption of only one layer of encryption. It achieves anonymity because the entry guard knows only the IP address of the client and the relay node but it does not know the IP address of the exit node. The exit node only knows the IP address of the relay node and destination node but does not know the IP address of the entry guard or the client. An example of how *TOR* works is shown in Fig. 6. In addition to opening connections to the Internet, users of *TOR* may connect to hidden services. The *TOR* network is a group of volunteer operated servers which helps improve security and privacy of Internet users. In our scheme, to protect the patients from traffic analysis attacks and to enhance the privacy, we have employed *TOR* at both server side and client side.

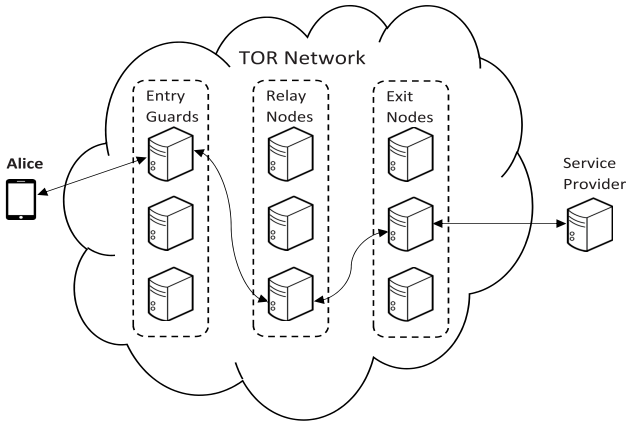

FIGURE 6. The working mechanism of TOR.

- On Server side: The *TOR* provides a program to run on the server side. This program can be configured to forward connections from a hidden service to a port on the local computer. In our proposed scheme, we must run two hidden services: one pointing to a program running *CSP*'s portion of the protocol and another pointing to a program running *RS*'s portion of the protocol. For testing purposes these programs may run on the same hardware, but they should run on different networks practically. Further details on *TOR* configuration can be found on the *TOR* project.[3]
- On Client side: *TOR* can be used as a proxy application to encrypt the Internet traffic and then hides it by bouncing through the series of computers around the world. Orbot[4] is an example of one of such applications. Orbot allows other applications on the same device to access hidden services. The test application can be connected to the hidden services through Orbot.

VI. SCHEME EVALUATION

A. PROOFS AND DISCUSSIONS

In this section, we will analyze the security of the proposed scheme and discuss how it achieves the privacy goals discussed in Section IV-B

1) INTRACTABLE PROBLEMS

The security of our proposed protocol is based upon the Elliptic-Curve Diffie-Hellman problem and the Inverse Bilinear Pairing Operation problem. The Elliptic-Curve Diffie-Hellman problem states that, given a point P on an elliptic curve the result of the Elliptic Curve Multiplication $C = aP$, it is computationally infeasible for someone to determine the value of a. The Inverse Bilinear Pairing Operation states that, given a bilinear pairing operation e and the result of a bilinear pairing operation $E = e(a, b)$, it is computationally infeasible for someone to determine the values of a or b. Therefore, any protocol whose security is based upon one or both of these problems is computationally secure.

2) ANONYMITY

The first feature that our protocol provides is anonymity for a user, *Alice*, from the service provider, *CSP*. This means that *Alice* can prove that she is authorized to use *CSP's* services without revealing her identity to *CSP*.

[3]https://www.torproject.org/docs/tor-hidden-service.html.en
[4]https://guardianproject.info/apps/orbot/

Proof: The values that *Alice* is attempting to conceal from *CSP* are r, t, x, and A. *CSP* is provided with the equations

$$a = Q.r \tag{21}$$
$$b = (P_1)^x.Q.r \tag{22}$$
$$d = r.A \tag{23}$$
$$c = e(d, RC.R.(P_2)^t) \tag{24}$$

CSP knows the values of a, b, d, e, Q, R, P_1, and P_2. To determine the value of r from the first equation, $(P_1)^x$ from the second equation, A from the third equation or $(P_2)^t$ from the fourth equation would violate the hardness of the Elliptic-Curve Diffie-Hellman problem. Therefore, it is computationally infeasible for *CSP* to determine the values of $(r, (P_1)^x, A, (P_2)^x)$, let alone (r, t, x, A). Similarly, given the fourth equation and the values of c and d, it is computationally infeasible to compute the second argument due to the hardness of the Inverse Bilinear Pairing Operation problem.

3) FORWARD UNLINKABILITY

Our protocol also provides forward unlinkability of transactions to *Alice*. This means that, unless *Alice* is placed on a revocation list, *CSP* does not have the ability to look at any two transactions and determine whether or not they were both performed by the same user.

Proof: The proposed scheme is forward-unlinkable because all values provided to *CSP* are dependent upon a random value, r. In order to connect requests, it is necessary to calculate the value of, r, which is computationally infeasible by the hardness of the Elliptic-Curve Diffie-Hellman and Inverse Bilinear Pairing Operation problems.

4) TRACEABILITY

Another feature provided by our protocol is traceability, which also allows revocation of credentials when a user is found to be abusing the system. This means that an authorized party has the ability to learn the identity of an abusive user and revoke their access to the system.

Proof: In order for the proposed scheme to be traceable, it must be possible for an authorized party to determine the user for a given authentication session. In this protocol, this is made possible through the combination of information stored by *RS* and *CSP*. During registration, *RS* stores the values of $((P_1)^x, A, t)$ for all users. In each registration session, *CSP* stores the values $a = Q.r$ and $d = A.r$. *Trent* can determine the value of A used in a given authentication session by iterating through *RS*'s list and therefore identify the user.

5) INTEGRITY AND ATTACK RESISTANCE

Our protocol is designed to guarantee the integrity of messages and protection against Man-in-the-Middle, eavesdropping, and replay attacks. This means that the recipient of a message can be confident that it came from the alleged sender and was not modified in transit.

Proof: The integrity of the protocol is dependent upon public key cryptography and knowledge of values shared over a secure channel. When setting up a group, *Trent* signs the group parameters using a private key whose public key was shared previously with *RS* and *CSP*. For all future communications, *RS* and *CSP* encrypt the data being sent using *Trent's* public key so that only he can decrypt it. Therefore, the integrity of all communications between *RS*, *Trent*, and *CSP* is guaranteed by public key cryptography. Communications between *Alice* and *RS* are assumed to occur over a secure channel and a failed session will result in *Alice* being unable to authenticate, revealing any failures in integrity. Communications between *Alice* and *CSP* rely upon information from these communications over a secure channel and a loss of integrity will result in a failure to authenticate. These protections secure the protocol against Man-in-the-Middle attacks and eavesdropping attacks.

The authentication stage of the protocol is protected against replay attacks due to the fact that both *Alice* and *CSP* generate a random parameter to include in the authentication calculation. An attacker that attempts to replay one side of the protocol will cause a failed authentication since one of these random numbers will be different. This difference in random numbers will cause Eq. (19) not to hold true as c will be calculated based upon the replayed value of *RC* but the *CSP* will be using the new value of *RC*, hence causing the *CSP* to reject the authentication attempt.

6) DENIAL OF SERVICE (DOS) ATTACKS

Same as all existing authentication schemes, our proposed system is vulnerable to *DoS* attacks due to the requirement for unlinkability. If requests could be linked, the detection of a *DoS* attack would be trivial and the future connection attempts from that user could be ignored. In order to identify an attacker and add them to a revocation list in a scheme providing unlinkability, anonymity must be stripped away from one or more users without being certain that their connection was part of the attack (since they could be legitimately using the service while the attack is taking place). Even if the attacker is added to the revocation list, several communications must be performed before the *CSP* tests for revocation. Therefore, a *DoS* attack attempting to tie up the *CSP*'s resources by repeating only the pre-test phases is still possible.

7) TRAFFIC ANALYSIS ATTACKS

We have used *TOR* to prevent our scheme against traffic analysis attacks to preserve the anonymity of the user. For anonymity *TOR* client selects the nodes to be used in circuits uniformly from the set of active routers so that the attacker cannot influence the selection path. Normally, an adversary can modify the communication in a variety of ways i.e., drop, alter, delay or add. However, for traffic analysis attacks, we only consider an adversary who wants to learn the source or destination of the communication. *TOR* can be insecure against an adversary who can observe the traffic as it

enters and leaves the *TOR* network. In that case, the adversary needs to have access to both entry and exit nodes to correlate the traffic using traffic analysis attack [52].

A method for path selection and specifying the degree of anonymity was proposed by Panchenko and Renner [53] using entropy based metric. A user can be deanonymized if the attacker owns both the entry and exit nodes of the user's circuit. The entropy $E(X)$ of a path selection metric and the corresponding degree of anonymity d from a sample path X can be calculated as follows [53]:

$$E(X) = -\sum_{0}^{N-1} p_i . \log_2(p_i) \tag{25}$$

$$d = 1 - \frac{E_{Max} - E(X)}{E_{Max}} = \frac{E(X)}{E_{Max}} \tag{26}$$

The number of the distinct entry and exit nodes combinations that occurred in the sample of paths X are denoted by N in Eq. (25), whereas the corresponding probability of the combination i is represented by p_i. E_{Max} used in Eq. (26) is the maximum entropy that can occur. Using $d = 1$ in a path selection method will choose any entry and exit nodes combination with the same probability. Hence, the attacker has no influence in the path selection process.

While the potential exists for an attacker to defeat the security provided by the TOR network in our proposed scheme, the result is that anonymity is lost due to factors outside the control of our protocol. Other protocols that do not implement protections using *TOR* or similar products will be vulnerable to such attacks by default.

B. COMPLEXITY ANALYSIS

In this section, we compare the complexity of the proposed scheme to several similar schemes from the literature. In selecting protocols for comparison, we used the criteria to narrow the selection of protocols:

- No devices are required by the protocol (smartcard, tamper-proof device, etc.)
- The protocol allows an unlimited number of authentications by a user without compromising privacy

Based on the criteria, the protocols selected for comparison are those by Djellalbia *et al.* [40], Kuzhalvaimozhi and Rao [27], Li *et al.* [41], Lin *et al.* [29], Liu *et al.* [42], and Sudarsono and Al Rasyid [43], which are referred to as Djellalbia, Kuzhalvaimozhi, Li, Lin, Liu, and Sudarsono respectively in the following Tables.

Table 2 compares the important features provided by the protocols studied. Subscripts provide further information for some values and are explained as follows:

1) No revocation algorithm is specified that allows the Service Manager to report abuse of services and terminate patient's access.
2) The user information shared between the patient and the Registration and Ticket Manager is never specified. It may be necessary to create a new user ID to update credentials.
3) The Group Manager can link user's public keys to their identity and trivially link user transactions together. Authors treat certificates as ''pseudo-secret'' since GM may not be trusted.
4) No use of public key infrastructure mentioned to protect communication integrity.
5) The patient's ''pseudo-identity"/private key is the hash of their true identity, a new private key requires a new identity.
6) The authors claim unlinkability of their scheme; however, the pseudonym sent to the Access Point is the same for each transaction and allows transactions to be linked.
7) It is never specified if a secure channel is used for registration. If not, it is possible to Man-in-the-Middle the protocol.
8) The GM knows every user's secret key. A malicious GM could impersonate any user.
9) The user's ID is "embedded in the key". Depending on how this occurs, credential update may require the user to create a new ID every time.
10) Only the signer creates random values in the signing phase, an attacker could replay a session with the same random values.

As shown in Table 2, our protocol is the only studied protocol that provides all of the desired features and protections against attacks. The protocol that comes closest is

TABLE 2. Comparison of features.

Features	Djellalbia [40]	Kuzhal-vaimozhi [27]	Li [41]	Lin [29]	Liu [42]	Sudarsono [43]	Our Protocol
Anonymity	✓	$\checkmark_3$	✓	✓	✓	✓	✓
Mutual Authentication	✓	×	✓	✓	✓	$\times_8$	✓
Forward Unlinkability	✓	$\checkmark_3$	$\times_5$	×	✓	✓	✓
Traceability	✓	✓	✓	✓	×	✓	✓
Revocation	$\times_1$	×	✓	✓	×	✓	✓
Efficient Credential Update	$\checkmark_2$	✓	✓	$\times_6$	✓	$\checkmark_9$	✓
Communication Integrity	✓	$\times_4$	✓	✓	✓	✓	✓
Resistance to Modification Attacks	✓	$\times_4$	✓	✓	✓	✓	✓
Resistance to MitM Attacks	✓	$\times_4$	✓	✓	$\checkmark_7$	✓	✓
Resistance to Replay Attacks	✓	×	✓	✓	✓	$\times_{10}$	✓

TABLE 3. Comparison of algorithmic complexities.

Protocols	Initialization	Registration	Authentication	Revocation
Djellalbia [40]	4 KG (RSA)	1 KG (RSA) 1 Ticket of Ticket Generation (undefined) 1 E_A 1 D_A	2 E_A 2 D_A 1 H 2 E_M 1 search through Access Tickets	Undefined
Kuzhalvaimozhi [27]	Choose parameters 1 KG	3 ECM 2 H 1 Zero-knowledge proof (undefined)	7 ECM 7 ECA 4 H 3 EP 1 E	Undefined
Li [41]	Choose parameters 1 KG	10 ECM 2 H 4 ECA 6 EP	3 ECM 2 ECA 2 EP 1 E_A 1 D_A 2 MAC	Undefined
Lin [29]	Choose parameters 1 KG	1 E_S 4 H 1 ECM 1 EP 1 KG	2 ECM 9 + 2*α H 1 E_S $n + 1/n$ identity based encryptions 2 EP 1 D_S	Undefined
Liu [42]	Choose parameters 2 KG	1 H 2 ECM 1 KG 2 EP	7 ECM 3 ECA 4 H 1 EP 2 E	Undefined
Sudarsono [43]	Choose parameters $2 * m + 2$ E	2 E	33 E $r + 7$ ECM $r + 9$ EP 2 H	1 E
Our Protocol	Choose parameters 1 KG 1 E 3 E_A 2 D_A	3 E 3 ECM 1 E_A 2 EP	1 E_A 2 E $r + 7$ ECM 2 EP 2 H	2 D_A $m + 1$ ECM

that of Li *et al.*, which lacks forward-unlinkability. It can be seen in [41] that the pseudoidentity generated in registration is provided as a part of every authentication packet used, making the transactions trivially linkable. Also, for several of the protocols listed, the issue of credential revocation is not explicitly addressed; however, if it appears possible that the protocol could provide credential revocation, the scheme is given credit for providing revocation as well.

In addition to this, the proposed method provides a description of how to use the *TOR* network to add a second layer of anonymity to the protocol. While all of the studied protocols provide theoretical anonymity, it is possible for an attacker monitoring network communications to trace packets back to users and therefore reveal their identities outside of the bounds of the protocol. Use of the TOR network is compatible with all of the protocols studied and is highly recommended in order to ensure user anonymity at all levels.

Next, we compare proposed protocol with selected protocols from literature for computational complexity. For each protocol, the four phases are decomposed into the operations necessary to complete the phase for a single user. In the following, m is used to denote the total number of users of the system, α denotes the number of patients assigned to a given physician, n is a variable set by the implementers of the protocol, and r is the number of revoked users. The complexities of each protocol are shown in Table 3. The symbols used in the table refer to the following operations:

- KG: Generate public/private keypair
- ECA: Elliptic-Curve Addition
- ECD: Elliptic-Curve Division
- ECM: Elliptic-Curve Multiplication
- H: Hash Operation
- E_S/D_S: Symmetric Key Encryption/Decryption
- E_A/D_A: Asymmetric Key Encryption/Decryption
- *EP*: Evaluation of pairing function

TABLE 4. Comparison of execution times.

Protocols	Initialization (ms)	Registration (ms)	Authentication (ms)	Revocation (ms)
Djellalbia [40]	4 KG (RSA)	100.79 + ToT Gen	150.62+A.T. Search	N/A
Kuzhalvaimozhi [27]	Choose parameters + 1 KG	121.25 + ZKP	625.64 + 7 ECA	N/A
Li [41]	Choose parameters + 1 KG	913.86 + 4 ECA	321.57 + 2 ECA + 2 MAC	N/A
Lin [29]	Choose parameters + 1 KG	185.47 + E_S + KG	385.56 + 29.24 α + $n + 1/n$ identity based enc + E_S + D_S	N/A
Liu [42]	Choose parameters + 2 KG	268.6 + KG	496.51 + 3 ECA	N/A
Sudarsono [43]	127.02 + Choose parameters + 127.02*m	127.02	3206 + 126.99*r	63.51
Our Protocol	217.15 + Choose parameters + 1 KG	493.64	582.05 + 30.67*r	67.59 + 30.67*m

TABLE 5. Communication overhead for each protocol.

	Initialization	Registration	Authentication	Revocation
Djellalbia [40]	0	2	10	N/A
Kuzhalvaimozhi [27]	1	3	1	N/A
Li [41]	1	4	2	N/A
Lin [29]	1	3	1+1/t	N/A
Liu [42]	1	2	2	N/A
Sudarsono [43]	0	2	2	1
Our Protocol	2	4	3	5

TABLE 6. Comparison of communication lengths per round based on the number of messages exchanged.

Protocols	Initialization (bits)	Registration (bits)	Authentication (bits)	Revocation (bits)
Djellalbia [40]	0	3081	6914	N/A
Kuzhalvaimozhi [27]	368	864	640	N/A
Li [41]	544	2592	1856	N/A
Lin [29]	378	768	1024+(320+n*1344)/n	N/A
Liu [42]	376	1952	1056	N/A
Sudarsono [43]	840+512*m	288	7045	32
Our Protocol	1480	576	2368	768*m+1024

- E: Exponentiation
- E_M: Modular Exponentiation
- P_C: Calculation of prime number

In this analysis, the complexities of looking up public keys or certificates in a database are not included as they are necessary to all of the described protocols and the complexity depends upon implementation details.

In an anonymous authentication scheme, the complexity of the authentication phase is the most significant as it is performed each time the user wishes to perform a transaction. As shown in Table 3, our protocol's authentication phase is more efficient than those proposed by Djellalbia *et al.* (which requires a search through all issued Authentication Tickets, which is likely several times greater than m), Lin *et al.* (which requires $n+1/n$ identity based encryption operations), and Sudarsono *et al.* (which requires $r + 7$ Elliptic Curve Multiplications and $r + 9$ evaluations of a pairing function as compared to our values of $r + 7$ and 2 respectively).

The proposed scheme is comparable to the proposal by Li *et al.* and that by Liu *et al.* These two protocols do not have a clearly defined revocation algorithm. Logically, it is impossible for a protocol to be unlinkable and not require performing some operation on every value in the revocation list or contacting a trusted third party during revocation. If the verifier could compute a value for each authentication and check for that value in the revocation list, transactions could be linked by the values computed. Therefore, tests for revocation must require at least r operations. This makes our protocol's authentication phase comparable in complexity to those of Li *et al.* and Liu *et al.*, since the primary difference between the complexity of our authentication phase and theirs is the r Elliptic Curve Multiplications that we must perform in order to test if a user is on the revocation list.

Kuzhalvaimozhi and Rao's protocol has a lower complexity than ours in Authentication and claims to provide anonymity and unlinkability. However, the Group Manager (GM) in their protocol can trivially de-anonymize a user and link transactions together. The scheme treats information usable for linkability as "pseudo-secret" since it could be shared by a corrupt or incompetent GM. Therefore, we do not consider this scheme to truly provide unlinkability. Also, this scheme provides no means of performing credential revocation. These two shortcomings make the protocol unusable for our targeted use case.

Table 4 shows the execution times provided in [42], which can be used to estimate the relative complexities of each protocol. One can see that our protocol is much faster than Sudarsono's protocol, which is the only other protocol that provides unlinkability, in the initialization and authentication phases. However, it is slower in the revocation phase. This is due to the fact that unlike Sodarsono's protocol, the new protocol treats the *CSP* as an untrusted entity.

Then, we compare the studied protocols based upon the number of one-way communications necessary for each phase of the protocol. The results are shown in Table 5. The value t is a special parameter of Lin's scheme that denotes the number of patient records to be bundled in a single transmission from the database to physicians. It is assumed for initialization that each publishing of a message requires a single transmission.

As shown in Tables 5 and 6, it is difficult to compare the communication complexities of the studied protocols without the knowledge of the revocation functions used. The only other protocol with a clearly defined revocation procedure, Sudarsono's protocol, requires fewer communications per phase than our protocol. However, as shown in Table 6, for the most commonly executed phase (Authentication), our protocol requires less than half the number of bits to be communicated for the messages exchanged during the authentication phase. The values used in Table 6 were calculated based upon the following assumptions.

1) For all protocols, the group public key is transmitted as a part of Initialization.
2) Hash functions have 256-bit outputs.
3) 160-bit public keys are used for the calculations on elliptic curves.
4) Message authentication codes are calculated as a 256-bit HMAC.
5) Identities when defined in protocols are stored as 32-bit values.
6) Each selection from a group of functions (hash, pairing, etc.) is stored as an 8-bit value.
7) The results of all exponentiations are stored as 256-bit values.
8) The generated random values are 32 bits.
9) Times and dates are stored as 64-bit values.
10) The result of a multiplication with a hash function output is stored in a 512-bit value (unless it is an elliptic curve multiplication).
11) Encryption is performed with a 256-bit block cipher.
12) All textual messages are 1024 bits in length.

VII. CONCLUSION

Protecting the privacy of patients is crucial to the success of smart cloud based healthcare applications. In this paper, we have presented the anonymous authentication scheme for smart cloud based healthcare applications. The proposed scheme preserves the privacy of patients when they access the services hosted on the cloud. The scheme utilizes rotating group signature scheme based on *ECC*. Due to smaller key sizes used in *ECC*, the security of the system can be easily scaled up by increasing the key size without affecting the computational complexity. The scheme adds an extra layer of protection against traffic analysis attacks by an eavesdropper by providing anonymity at the network layer by employing *TOR*. The scheme protects patients' sensitive data from an eavesdropper and untrusted cloud servers. One salient feature of our scheme is that the medical application or service providers cannot reveal the identity of the patient hence protecting the privacy. In this paper, we have designed a practical system which is secure and efficient. The proposed authentication scheme ensures that the patients can consume services without revealing their identity at the time of consumption or retrospectively.

REFERENCES

[1] A. Martinez-Balleste, P. A. Perez-Martinez, and A. Solanas, "The pursuit of citizens' privacy: A privacy-aware smart city is possible," *IEEE Commun. Mag.*, vol. 51, no. 6, pp. 136–141, Jun. 2013.

[2] D. Aranki, G. Kurillo, P. Yan, D. M. Liebovitz, and R. Bajcsy, "Real-time tele-monitoring of patients with chronic heart-failure using a smartphone: Lessons learned," *IEEE Trans. Affective Comput.*, vol. 7, no. 3, pp. 206–219, Jul./Sep. 2016.

[3] Z. Xiao and Y. Xiao, "Security and privacy in cloud computing," *IEEE Commun. Surveys Tuts.*, vol. 15, no. 2, pp. 843–859, 2nd Quart., 2013.

[4] D. Ding, M. Conti, and A. Solanas, "A smart health application and its related privacy issues," in *Proc. Smart City Secur. Privacy Workshop (SCSP-W)*, Apr. 2016, pp. 1–5.

[5] P. Gope and T. Hwang, "Untraceable sensor movement in distributed IoT infrastructure," *IEEE Sensors J.*, vol. 15, no. 9, pp. 5340–5348, Sep. 2015.

[6] X. Su *et al.*, "Privacy as a service: Protecting the individual in healthcare data processing," *Computer*, vol. 49, no. 11, pp. 49–59, 2016.

[7] W. Lei, Y. Li, Y. Sang, and H. Shen, "A secure anonymous authentication scheme for electronic medical records system," in *Proc. 13th Int. Conf. e-Bus. Eng.*, Nov. 2016, pp. 48–55.

[8] V. Sucasas, G. Mantas, A. Radwan, and J. Rodriguez, "An OAuth2-based protocol with strong user privacy preservation for smart city mobile e-Health apps," in *Proc. IEEE Int. Conf. Commun.*, May 2016, pp. 1–6.

[9] R. Fernando, R. Ranchal, B. An, L. B. Othman, and B. Bhargava, "Consumer oriented privacy preserving access control for electronic health records in the cloud," in *Proc. IEEE 9th Int. Conf. Cloud Comput.*, Jun./Jul. 2016, pp. 608–615.

[10] A. Mehmood, I. Natgunanathan, Y. Xiang, G. Hua, and S. Guo, "Protection of big data privacy," *IEEE Access*, vol. 4, pp. 1821–1834, 2016.

[11] H. Xiong, J. Tao, and C. Yuan, "Enabling telecare medical information systems with strong authentication and anonymity," *IEEE Access*, vol. 5, pp. 5648–5661, 2017.

[12] X. Li, S. Tang, L. Xu, H. Wang, and J. Chen, "Two-factor data access control with efficient revocation for multi-authority cloud storage system," *IEEE Access*, vol. 5, pp. 393–405, 2017.

[13] J. Zhu and J. Ma, "A new authentication scheme with anonymity for wireless environments," *IEEE Trans. Consum. Electron.*, vol. 50, no. 1, pp. 231–235, Feb. 2004.

[14] G. Horn and B. Preneel, "Authentication and payment in future mobile systems," in *Proc. Comput. Secur. (ESORICS)*, 1998, pp. 277–293.

[15] C. Yang, W. Ma, and X. Wang, "Novel remote user authentication scheme using bilinear pairings," in *Autonomic and Trusted Computing* (Lecture Notes in Computer Science) vol. 4610. Berlin, Germany: Springer, 2007, pp. 306–312.

[16] P. E. Abi-Char, A. Mhamed, and B. El-Hassan, "A fast and secure elliptic curve based authenticated key agreement protocol for low power mobile communications," in *Proc. Int. Conf. Next Gener. Mobile Appl., Services Technol.*, 2007, pp. 235–240.

[17] L. Zhang, S. Tang, and H. Luo, "Elliptic curve cryptography-based authentication with identity protection for smart grids," *PLoS ONE*, vol. 11, no. 3, p. e0151253, 2016.

[18] Y.-M. Tseng, T.-Y. Wu, and J.-D. Wu, "A mutual authentication and key exchange scheme from bilinear pairings for low power computing devices," in *Proc. 31st Annu. Int. Conf. Comput. Softw. Appl.*, vol. 2, Jul. 2007, pp. 700–710.
[19] J.-H. Yang and C.-C. Chang, "An ID-based remote mutual authentication with key agreement scheme for mobile devices on elliptic curve cryptosystem," *Comput. Secur.*, vol. 28, nos. 3–4, pp. 138–143, Jun. 2009.
[20] X. Cao, X. Zeng, W. Kou, and L. Hu, "Identity-based anonymous remote authentication for value-added services in mobile networks," *IEEE Trans. Veh. Technol.*, vol. 58, no. 7, pp. 3508–3517, Sep. 2009.
[21] R. Lu, X. Lin, H. Zhu, P. H. Ho, and X. Shen, "A novel anonymous mutual authentication protocol with provable link-layer location privacy," *IEEE Trans. Veh. Technol.*, vol. 58, no. 3, pp. 1454–1466, Mar. 2009.
[22] I. Teranishi, J. Furukawa, and K. Sako, "*K*-times anonymous authentication," *IEICE Trans. Fundam. Electron., Commun. Comput. Sci.*, vol. E92-A, no. 1, pp. 147–165, Jan. 2009.
[23] K. G. Paterson, "ID-based signatures from pairings on elliptic curves," *Electron. Lett.*, vol. 38, no. 18, pp. 1025–1026, Aug. 2002.
[24] F. Hess, "Efficient identity based signature schemes based on pairings," in *Proc. Int. Workshop Sel. Areas Cryptogr.*, 2002, pp. 310–324.
[25] J. C. Choon and J. H. Cheon, "An identity-based signature from gap Diffie–Hellman groups," in *Proc. Int. Workshop Public Key Cryptogr.*, 2003, pp. 18–30.
[26] D. Wang, H. Cheng, D. He, and P. Wang, "On the challenges in designing identity-based privacy-preserving authentication schemes for mobile devices," *IEEE Syst. J.*, vol. 12, no. 1, pp. 916–925, Mar. 2016.
[27] S. Kuzhalvaimozhi and G. R. Rao, "An efficient scheme for anonymous authentication using identity based group signature," in *Proc. 4th Int. Conf. Adv. Recent Technol. Commun. Comput.*, 2012, pp. 139–142.
[28] S. Gupta and V. Gupta, "Revocable key identity based cryptography without key escrow problem," in *Proc. Int. Conf. Comput., Commun. Autom.*, 2016, pp. 443–448.
[29] X. Lin, R. Lu, X. Shen, Y. Nemoto, and N. Kato, "Sage: A strong privacy-preserving scheme against global eavesdropping for eHealth systems," *IEEE J. Sel. Areas Commun.*, vol. 27, no. 4, pp. 365–378, May 2009.
[30] J. Sun, Y. Fang, and X. Zhu, "Privacy and emergency response in e-healthcare leveraging wireless body sensor networks," *IEEE Wireless Commun.*, vol. 17, no. 1, pp. 66–73, Feb. 2010.
[31] J. Sun, X. Zhu, C. Zhang, and Y. Fang, "HCPP: Cryptography based secure EHR system for patient privacy and emergency healthcare," in *Proc. 31st Int. Conf. Distrib. Comput. Syst.*, Jun. 2011, pp. 373–382.
[32] T.-F. Lee "An efficient chaotic maps-based authentication and key agreement scheme using smartcards for telecare medicine information systems," *J. Med. Syst.*, vol. 37, no. 6, p. 9985, Dec. 2013.
[33] T.-F. Lee, "Verifier-based three-party authentication schemes using extended chaotic maps for data exchange in telecare medicine information systems," *Comput. Methods Programs Biomed.*, vol. 117, no. 3, pp. 464–472, 2014.
[34] X. Xu, P. Zhu, Q. Wen, Z. Jin, H. Zhang, and L. He, "A secure and efficient authentication and key agreement scheme based on ECC for telecare medicine information system," *J. Med. Syst.*, vol. 38, no. 1, p. 120, Jan. 2014.
[35] F. Wen and D. Guo, "An improved anonymous authentication scheme for telecare medical information systems," *J. Med. Syst.*, vol. 38, no. 5, pp. 26–38, May 2014.
[36] D. Mishra, S. Mukhopadhyay, S. Kumari, M. K. Khan, and A. Chaturvedi, "Security enhancement of a biometric based authentication scheme for telecare medicine information systems with nonce," *J. Med. Syst.*, vol. 38, no. 5, pp. 41–53, Apr. 2014.
[37] Z. Tan, "A user anonymity preserving three-factor authentication scheme for telecare medicine information systems," *J. Med. Syst.*, vol. 38, no. 3, pp. 16–26, Apr. 2014.
[38] H. Arshad and M. Nikooghadam, "Three-factor anonymous authentication and key agreement scheme for telecare medicine information systems," *J. Med. Syst.*, vol. 38, no. 12, pp. 136–149, Dec. 2014.
[39] R. Amin and G. P. Biswas, "A secure three-factor user authentication and key agreement protocol for TMIS with user anonymity," *J. Med. Syst.*, vol. 39, no. 8, pp. 78–98, Aug. 2015.
[40] A. Djellalbia, N. Badache, S. Benmeziane, and S. Bensimessaoud, "Anonymous authentication scheme in e-Health Cloud environment," in *Proc. 11th Int. Conf. Internet Technol. Secured Trans.*, Dec. 2016, pp. 47–52.
[41] T. Li, Y. Zheng, and T. Zhou, "Efficient anonymous authenticated key agreement scheme for wireless body area networks," *Secur. Commun. Netw.*, vol. 2017, Oct. 2017, Art. no. 4167549.
[42] J. Liu, Z. Zhang, X. Chen, and K. S. Kwak, "Certificateless remote anonymous authentication schemes for wireless body area networks," *IEEE Trans. Parallel Distrib. Syst.*, vol. 25, no. 2, pp. 332–342, Feb. 2014.
[43] A. Sudarsono and M. U. H. Al Rasyid, "An anonymous authentication system in wireless networks using verifier-local revocation group signature scheme," in *Proc. Int. Seminar Intell. Technol. Appl.*, Jul. 2016, pp. 49–54.
[44] X. Zhu, S. Jiang, L. Wang, and H. Li, "Efficient privacy-preserving authentication for vehicular ad hoc networks," *IEEE Trans. Veh. Technol.*, vol. 63, no. 2, pp. 907–919, Feb. 2014.
[45] X. Zhu, S. Jiang, L. Wang, H. Li, W. Zhang, and Z. Li, "Privacy-preserving authentication based on group signature for VANETs," in *Proc. IEEE Globecom Workshops*, Dec. 2013, pp. 4609–4614.
[46] J. Petit, F. Schaub, M. Feiri, and F. Kargl, "Pseudonym schemes in vehicular networks: A survey," *IEEE Commun. Surveys Tuts.*, vol. 17, no. 1, pp. 228–255, 1st Quart., 2015.
[47] W. Feng, Z. Yan, and H. Xie, "Anonymous authentication on trust in pervasive social networking based on group signature," *IEEE Access*, vol. 5, pp. 6236–6246, 2017.
[48] S. G. Choi, K. Park, and M. Yung, "Short traceable signatures based on bilinear pairings," in *Proc. Adv. Inf. Comput. Secur. (IWSEC)*, vol. 6, Oct. 2006, pp. 88–103.
[49] V. S. Miller, "Use of elliptic curves in cryptography," in *Proc. Conf. Theory Appl. Cryptograph. Techn.*, 1985, pp. 417–426.
[50] D. Chaum and E. Van Heyst, "Group signatures," in *Proc. Adv. Cryptol.*, 1991, pp. 257–265.
[51] R. Dingledine, N. Mathewson, and P. Syverson, "Tor: The second-generation onion router," Naval Res. Lab., Washington, DC, USA, Tech. Rep., 2004.
[52] A. Panchenko, F. Lanze, and T. Engel, "Improving performance and anonymity in the Tor network," in *Proc. Conf. Perform. Comput. Commun.*, 2012, pp. 1–10.
[53] A. Panchenko and J. Renner, "Path selection metrics for performance-improved onion routing," in *Proc. Int. Symp. Appl. Internet*, 2009, pp. 114–120.

ABID MEHMOOD received the bachelor's degree in computer science from the COMSATS Institute of Information Technology, Pakistan, in 2006, and the master's degree in information technology from the University of Ballarat, Australia, in 2009. He is currently pursuing the Ph.D. degree with Deakin University, Australia.

IYNKARAN NATGUNANATHAN (M'15) received the B.Sc.Eng. degree (Hons.) in electronics and telecommunication engineering from the University of Moratuwa, Katubedda, Sri Lanka, in 2007, and the Ph.D. degree from Deakin University, Burwood, VIC, Australia, in 2012. From 2006 to 2008, he was a Software Engineer with Millennium Information Technology (Pvt) Ltd., Malabe, Sri Lanka. He is currently a Research Fellow with the School of Information Technology, Deakin University. His research interests include digital watermarking, audio and image processing, telecommunication, and robotics.

YONG XIANG (SM'12) received the Ph.D. degree in electrical and electronic engineering from The University of Melbourne, Australia. He is currently a Professor and the Director of the Artificial Intelligence and Image Processing Research Cluster, School of Information Technology, Deakin University, Australia. His research interests include information security and privacy, signal and image processing, data analytics and machine intelligence, and Internet of Things. He has published two monographs, over 100 refereed journal articles, and numerous conference papers in these areas. He has served as the Program Chair, the TPC Chair, the Symposium Chair, and the Session Chair for a number of international conferences. He is an Associate Editor of the IEEE SIGNAL PROCESSING LETTERS and the IEEE ACCESS.

HOWARD POSTON received the B.Sc. degree in computer engineering and the B.E.E. degree in electrical engineering from the University of Dayton in 2013, and the M.S. degree in cyber operations from the Air Force Institute of Technology in 2015. His research interests include cryptography and analysis of malware.

YUSHU ZHANG (M'17) received the Ph.D. degree from the College of Computer Science, Chongqing University, China, in 2014. From 2015 and 2017, he was an Associate Professor with the School of Electronics and Information Engineering, Southwest University, China. He held various research positions at the City University of Hong Kong and the University of Macau. He is currently an Alfred Deakin Post-Doctoral Research Fellow with the School of Information Technology, Deakin University, Australia. His research interests include multimedia security, compressive sensing security, cloud computing, and big data security. He has published over 70 refereed journal articles and conference papers in these areas.