EasyChair Preprint

№ 855

Designing a Secure and Privacy-Preserving Data Collection (SPDC) Framework for Collecting Data from Mobile Patients

Tahani Aljohani and Ning Zhang

March 26, 2019

Designing a Secure and Privacy-Preserving Data Collection (SPDC) Framework for Collecting Data from Mobile Patients

Tahani Aljohani
tahani.aljohani@postgrad.manchester.ac.uk
University of Manchester
UK, Manchester

Ning Zhang
ning.zhang-2@manchester.ac.uk
University of Manchester
UK, Manchester

ABSTRACT

Third-party based mobile health monitoring systems are vulnerable to threats not only imposed by outsiders but also authorized insiders, e.g. employees of the third-party service provider. This paper examines issues in this context and proposes a novel framework, called a secure and ID privacy preserving framework (SPDC), to facilitate secure and ID privacy-preserving data collections from remote patients. The framework has a number of properties: (i) it supports three different modes of data collections and different treatment of data with two levels of urgency, (ii) it protects the confidentiality and authenticity of data being collected, (iii) it preserves patients' ID privacy against external entities as well as entities that are involved in facilitating the data collections, (iv) it uses distributed load-sharing so that no single entity is wholly responsible for the task of data collections. The ideas used in achieving these properties include the use of structured use of multiple data collection servers and multi-level hierarchical pseudonyms to hide patients' IDs as well as their communication patterns, and the separation of duties and pseudonym and blind-token based authentication methods so that patients' can be identified and authenticated without revealing their real IDs. Security analysis against design requirements and anonymity evaluation using entropy method are carried out to demonstrate that the framework can resists attacks on data security and protects the patients' IDs from being revealed to unauthorized entities.

KEYWORDS

IoT, e-Health, Security, Privacy, Pseudonym, Anonymity, Authentication

ACM Reference Format:
Tahani Aljohani and Ning Zhang. 2019. Designing a Secure and Privacy-Preserving Data Collection (SPDC) Framework for Collecting Data from Mobile Patients . In *Proceedings of* . , 10 pages.

1 INTRODUCTION

The Internet of Things (IoTs) can be defined as a network of interconnected objects by means of information and communication technologies to create intelligent systems. These objects harvest information from the environment by sensing, interacting among each other, exchanging information and using the Internet standard protocols for delivering services and applications. These features (i.e. sensing and communicating) result in the creation of many innovative systems. One of these systems is the Mobile Patient Monitoring (MPM) system [1, 2] .

The main function of an MPM system is to provide remote patient health monitoring services anywhere and anytime. A conventional MPM system consists of some wearable devices worn by a patient, a mobile device carried by the patient and a backend server owned or managed by a healthcare provider. The wearable devices (e.g. wristbands, health patch) are used to measure some health data (e.g. heart rate, blood pressure) from the patient's body, and send the collected data to the remote server via the mobile device. The collected data may be further processed and used for clinical decision making. These collected data are collectively called Patient Generated Health Data (PGHD) [3, 4]. Future MPM systems are anticipated to be built on third-party owned infrastructures with more resourceful storage and data processing capabilities such as those by Microsoft, Amazon, and Google [5]. This is because on-premise infrastructures, which most of the healthcare providers are relying on today, may not be able to handle the volumes of PGHD generated by wearable devices. It is expected that, by 2021, more than 222 million wearable devices may be poured into the market and 3 in 5 patients may use remote monitoring services. These devices can generate PGHD at high (e.g. every 5 minutes) frequencies, leading to massive volumes of PGHD being generated [6].

When patients' PGHD are handled by a third-party service provider, a number of security and privacy concerns arise. These concerns include breach of data confidentiality and authenticity and unauthorized exposure of patients' real IDs, thus compromising their privacy. Existing data collection systems [7–17] do protect data confidentiality and preserve patients' ID privacy. However, there are limitations in these systems and proposals in terms of supporting scalability and protection against inference of patients' real IDs via observing their communication patterns and against privacy threats imposed by authorized insiders.

This paper reports our on-going work on overcoming these limitations by presenting the design of a novel framework, the SPDC framework, for collecting data from mobile patients. The framework consists of a system architecture, two sets of methods and protocols. This paper describes the system architecture and the methods. The protocol part will be described in a future paper. The SPDC framework has a number of properties. Firstly, it supports three different modes of data collections and different treatment of data with two levels of urgency. The three data collection modes are periodical data collection, on-demand data collection and event-driven data collection mode. The two levels of urgency are normal data and emergency data and the latter class of data is treated with a higher priority during a data delivery process by the data collection system. Secondly, it protects the confidentiality and authenticity of

data being collected and it preserves patients' ID privacy against external entities as well as entities that are involved in facilitating the data collections. Furthermore, it distributes processing load across multiple entities in the system so that no single entity is wholly responsible for the tasks of data collections and the provisioning of the security and privacy- preserving properties, thus improving system scalability. The ideas used in achieving these properties also include the use of multiple data collection servers and multi-level hierarchical pseudonyms to hide patients' IDs as well as their communication patterns, and the separation of duties and pseudonym and blind-token based authentication methods so that patients' can be identified and authenticated, ensuring authorized access of the data collection service, without revealing their real IDs.

In detail, the paper is structured as follows. Section 2 describes a use-case scenario, discusses security and privacy threats based on the scenario and specifies a set of requirements for the design of a secure and privacy-preserving data collection system and, based on the requirements, Section 3 critically analyses related work, so as to identify areas for improvements. Section 4 presents the ideas used in the design of SPDC and highlights its core components, i.e. the SPDC architecture and two sets of methods, respectively, for pseudonym generation and linkage and anonymous authentication based on the pseudonyms. Section 5 describes the pseudonym generation and linkage methods. Section 6 describes the anonymous authentication methods. Section 7 analyses the design against the requirements and its privacy preservation property using an entropy-based method. Finally, Section 8 concludes the paper and outlines future work.

2 THREATS ANALYSIS AND REQUIREMENT SPECIFICATION

2.1 Threat Analysis

To understand the threats to data security and ID privacy, we here carry out a threat analysis based on a use-case scenario. Assuming that a patient, Alice, suffers from a heart condition that needs to be monitored very frequently. Alice's healthcare provider does the monitoring via the use of wearable devices. These devices collect Alice's health data in real-time and send the data to a data collection server via her mobile device. This data collection process is performed regularly, say every 5 minutes. The server is managed or owned by a third party and can be accessed by Alice's healthcare provider. If the collected data indicates that Alice may have a health problem or an urgent condition, the server will notify the healthcare provider which may take further actions, e.g. sending instructions to Alice via her mobile phone or to request further information from her wearable devices.

There are a number of security and privacy concerns or threats in the above data collection process.Firstly, an adversary may try to impersonate Alice to gain unauthorized access to the data collection system. This could be for investigating what is attackable in the system. The adversary may also try to impersonate the data collection server to gain unauthorized access to Alice's health data or login credentials. Such impersonation attacks are sometimes also referred to as identity theft in literature. Secondly, if Alice's data is not protected properly during transit, and if an adversary gets hold of her data, e.g. by eavesdropping the channel, Alice's private and sensitive information, such as Alice' ID, her medical conditions and on what medication she is on, may be revealed or exposed to unauthorized entities. The exposure of private or sensitive information about patients could have serious consequences on them. For example, if a pharmaceutical company or its agent gets to know such information, it may target at the patients to persuade them to buy their products. If such information is revealed to potential employers, the patients may be deprived of job opportunities, etc. It should be emphasized that even if Alice's real ID is hidden or disguised by using a pseudonym during the data collection process, her real ID may still be revealed by inference. For example, if only Alice accesses the data collection server at this particular frequency and if every data collection from Alice carries the same pseudonym, then the data collection server would be able to learn that these data are collected from the same patient. By collating this information from information elsewhere, the data collection server may be able to work out the real ID of Alice. In addition, there are risks of active attacks. For example, Alice's data may be delayed, replayed or even modified during transit. An attacker may even forge data as if it is from Alice. Such attacks, if successful, may lead to the break-in of the data collection system or compromise of the integrity of data stored in the system, which can cause serious consequences to patients using the system.

2.2 Design Requirements

To design a data collection system that could accomplish the task of collecting health data from remote patients anywhere anytime in a secure, scalable and ID privacy-preserving manner, in the following we specify a set of requirements. The requirements can be classified into functional (F), ID privacy preservation (P), security (S1, S2, S3) and efficiency (E1, E2). The security requirements are for providing the assurance that the data collection service should only be accessed by authorized patients and that the confidentiality and authenticity of the data being collected be preserved during transit. ID privacy preservation requirement assures that the real ID of a patient should not be revealed during a data collection process. The efficiency requirements are aimed at accomplishing these functional, security and ID privacy preservation requirements with as less overheads as possible and in a scalable manner. The details of these requirements are as follows.

(F) Support various modes of data collections: The collection of data should be such that various modes of collections are supported. These are periodical, command-driven and event-driven data collection modes.

(P) Preserve patient's ID privacy: To satisfy this requirement, the following two requirements (P1,P2) should be satisfied. (P1) Provide patient's ID anonymity. (P2) Make different uploadings (one session may have a number of uploadings) and a pattern of interaction by the same patient unlinkable.

(S) Ensure entity authentication and data security: To provide these assurances, the following three security requirements (S1,S2,S3) should be satisfied. (S1) Support mutual entity authentication. Entity authentication ensures that a communicating entity is indeed whom it claims to be. This requirement is to counter impersonation attacks. This requirement should be satisfied without compromising patients' ID privacy. (S2) Provide end-to-end

data authenticity. Data authenticity assures that data are indeed from the claimed source and that it is exactly the same as what has been sent by the original sender. This requirement is to counter tempering, replay, or forgery attacks on data in transit. (S3) Provide end-to-end data confidentiality. Confidentiality protects data against unauthorized disclosure. This requirement is to protect against unauthorized access to data in transit.

(E) Make the design as efficient and as scalable as possible: This encompasses requirements (E1,E2). (E1) Minimize computational and communication overheads. Achieving the above-mentioned security and privacy properties impose additional computational and communication costs, and such overhead costs should be as low as possible. (E2) Support scalability. The data collection system should be scalable in that as the number of patients and/or the data generated by patients increases, the overheads costs imposed on any single entity in the system should not increase sharply.

Table 1: Related Works Analysis

Related Works	F	S1	S2	S3	P1	P2	E1	E2
[7]	-C	F	F	F	F	F	F	T
[8]	-C	F	T	T	F	F	F	T
[9]	-C	F	T	T	F	F	F	T
[10]	-C	F	T	T	F	F	F	T
[11]	T	T	T	T	T	F	F	T
[12]	T	T	T	T	F	F	F	T
[13]	-C	T	T	T	T	F	F	F
[14]	-C	T	T	T	T	F	F	F
[15]	-C	T	T	T	T	F	F	F
[16]	-C	T	T	T	T	F	F	F
[17]	-C	T	T	T	T	F	F	F

3 RELATED WORKS

This section critically analyses related work, i.e. data collection systems in healthcare arena, against the requirements specified above so as to identify knowledge gaps.

Existing data collection systems can largely be classified into two groups, centralized and distributed systems. In the centralized data collection systems [12–17], patients upload their health data onto a centralized data collection server where data are accessed by a healthcare provider. These systems typically rely on the use of a single entity (a single server or a single architectural component) to perform the tasks required to facilitate data collections. These tasks include: i) manage crypto key generations and distributions, ii) authenticate service users (patients), (iii) send commands and receive data from patients, (iv) process received data, e.g. decrypt the data, identify data owners and link multiple pieces of data from the same patients, etc. Using a single entity to perform all these tasks may create a single point . The relying on a single entity may create a single point of performance and security bottleneck, making the system less scalable. Research for these systems is largely focused on how to protect the confidentiality of data and how to preserve the anonymity of patients. More specifically, Merza et al. [12]proposed a secure end-to-end protocol for collecting data from the patient. However, the protocol uses the same patient's ID in each data uploading. In [14] the authors proposed a secure and privacy-preserving framework and the framework uses an identity-based cryptosystem to achieve security and privacy. For each patient, a number of unlinkable pseudonym IDs are issued. For each pseudonym ID, a pair of keys (i.e. public and a private one) are generated. When a patient wants to upload his/her data onto the healthcare provider server, the patient encrypts the data and tags it with one of the pseudonym IDs. By using a different pseudonym for each communication with the server, it is hard for an external attacker to link the multiple uploading performed by the same patient. Lin et al. [15] proposed a privacy-preserving scheme which achieves content and contextual privacy. The content privacy achieved by encrypting the patient's data before uploading them to the server, while the contextual privacy is achieved by breaking the relationship between the patient and his/her physician. To do this, the healthcare provider delivers the patient's data to his/her physician. In [16] the authors proposed a privacy-preserving protocol known as PEC to enable patients in life-threatening situations to report their health data to the nearby physician in a secure manner. In summary, these research efforts [12–17] are largely on achieving end-to-end security of data in transit. They have given little attention to issues such as how to handle a large number of patients and/or a large volume of data or how to make the system scalable in the presence of big data collections. In addition, these systems only support data flow from patients to the data collection server, not command flow from the healthcare provider to the patients.

With regard to the distributed data collection systems [7–11], these systems provide distributed servers across a geographical area. The patient chooses any one of these servers to upload his/her data on. These systems are more scalable and less secure. In [7] the authors acknowledged the importance of security and privacy, but they did not provide any security and privacy means to protect the confidentiality of the patient's data and the privacy of the patient's ID. The works [8–10] have proposed a method to preserve the patient's data by using an encryption method. In [11], the authors suggested a secure data collection protocol which supports entity authentication, data authenticity and confidentiality and protects the patient's ID.

The contributions made in the above mentioned existing systems or proposals as against our design requirements have been summarized in Table 1. From the table, it can be seen that, although a great deal of research has been carried out in achieving secure and ID privacy-preserving data collections from remote patients, none of the existing solutions has satisfied all the requirements specified in Section 2. To improve on the existing works, we have designed the SPDC Framework. The legends which are used in the table can be explained as follows. (-C) means the design does not support bi-directional communications,(T) the design supports the requirement and (F) the design does not support the requirement.

4 SPDC FRAMEWORK

This section first highlights the high-level ideas used in the design of SPDC, and then describes the SPDC architecture and the methods.

4.1 SPDC High-Level Ideas

The design of the SPDC framework has made use of the following ideas.

Idea 1. Use multiple data collection modes and differential data delivery approach to satisfying the functional requirement (F): In a healthcare context, there are multiple reasons for collecting a patient's data remotely. For example, it could be for monitoring the patient's medical condition, part of a clinical research, for administrating some medication, part of a treatment process, or for notifying a hospital of an urgent medical condition, part of the emergence service, etc. For these different usecase scenarios, how, when and how often the data should be collected may differ. For example, for the monitoring purpose, it is more likely that the data should be collected regularly or periodically. On the other hand, for the emergency service, the collection is more likely to be event driven. Based on these considerations, we have identified three data collection modes: (i) periodical data collection mode with which data are collected periodically with a predefined interval, (or frequency), (ii) command-driven data collection mode with which data are only collected upon the receipt of a command from the healthcare provider, and (iii) event-driven data collection mode with which data are collected upon the triggering of an event at the patient side.

Idea 2. Use pseudonyms and multiple data collection servers to satisfy ID privacy preservation requirement (P): For each patient, data are collected by (or uploaded onto) his/her health provider server via data collection servers (the justifications for the use of the data collection servers are to be explained shortly). This data collection process is typically executed repeatedly or periodically. Preserving the patient's ID privacy requires that (a) each such data uploading be labelled or identified by an artificial ID (i.e. a pseudonym), rather than the patient's real ID, and (b) any linkage among the different uploadings from, or patterns of communications exposed by, the same patient be hidden. Justifications for (a) is intuitive, as data to be delivered from a patient's device to the health service provider's server should not contain any identifying information (i.e. information that may be used to identify the patient). Pseudonyms should be used to label the data so that authorized entities can link the multiple packets of data from the same patient. This authorized data linkage is necessary for clinical purposes and also for reducing bandwidth costs (to be explained shortly).

Justifications for (b) can be explained by using an example: if a patient, Alice, always uploads her data onto a data collection server (B), at 1pm every day (say for blood pressure) and every 10 minutes (say for heart rhythm), then even if the patient uses a pseudonym, by observing the pattern of uploading (i.e. 1 pm and every 10 minutes), unauthorized entity (e.g. data collection server (B)) may be able to infer that all the data uploaded at this time may belong to the same patient. To reduce such risks (and also for the sake of making the system more scalable as to be explained shortly), multiple data collection servers are used in the design of the SPDC architecture. The patient can select how many servers s/he uses, and when to use which server, to upload their PGHD, in a given time period. In this way, the patterns of uploading performed by a given patient can better be hidden. The more servers a patient chooses to use and the more random the selection of the servers, the harder it is to infer the real ID of the patient.

Idea 3. Use anonymous authentication to satisfy the security requirements (S1): To ensure authorized use of the SPDC data collection service (i.e. satisfying S1), patient should be identified and authenticated before they are allowed to access the service and this should be achieved without compromising the patient's ID privacy. This means that patient identification and authentication should be carried out in an anonymous manner. In addition, as mentioned above, to reduce the risk of ID inference based on uploading patterns, a patient may choose to use (e.g. by randomly selecting) one or more data collection servers to upload their data. The authentication solution designed should support the anonymous authentication of a patient with multiple data collection servers, bearing in mind that the set of multiple servers used by a patient, both in terms of the number in the set and the server identities (which servers), may be selected dynamically by the patient. To support such an authentication service in a seamless and scalable manner (to be further discussed below), for each patient, we denote one data collection server as the patient's home server and the rest as the patient's foreign servers. We have designed two anonymous authentication methods , a pseudonym certificate based authentication method for the authentication of a user to his/her home data collection server and a blind token based authentication method for the authentication of the user to his/her foreign data collection servers.

Idea 4. Use data encryption and signing to satisfy the security requirements (S2, S3): Before a patient uploads his/her data to any of the data collection servers, the patient should first sign the data using his/her privet key and then encrypt the data and the signature using symmetrical key encryption. The key is only known to the patient and healthcare provider server.

Idea 5. Use data aggregation and the principles of distributed load-sharing and separation of duties to satisfy the efficiency and scalability requirements (E1 and E2): Patients' data are classified into two categories, normal data and urgent data. Normal data are aggregated by data collection servers before being forwarded to the final destination, the healthcare provider server, and this is to save bandwidth costs. Urgent data are forwarded by the first data collection server reached by the data to the healthcare provider server without further processing. This is to reduce any intermediate delays so that data could reach to the destination as soon as possible. The authentication credentials used by each patient are of two categories, one for the authentication of patients with their respective home data collection servers and the other for the authentication of patients with their respective foreign data collection servers. The former are issued by the healthcare provider, while the latter are issued by the patients and signed blindly by their respective home data collection servers.

4.2 The SPDC Architecture

The SPDC system architecture is shown in Figure 2. It consists of a patient's mobile device, multiple data collection servers structured as follows, one home data collection server (hDCsrv), foreign data collection servers (fDCsrvs) and healthcare provider server (HCPsrv).

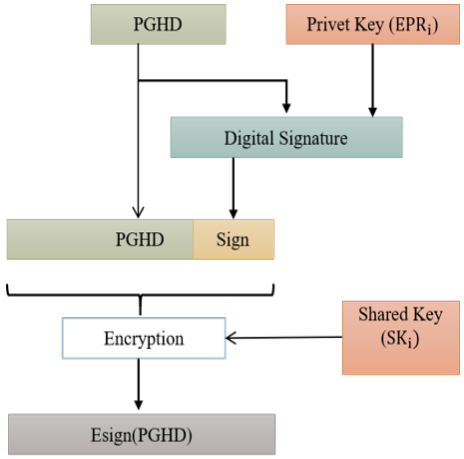

Figure 1: Sign-then-Encrypt (StE) Method

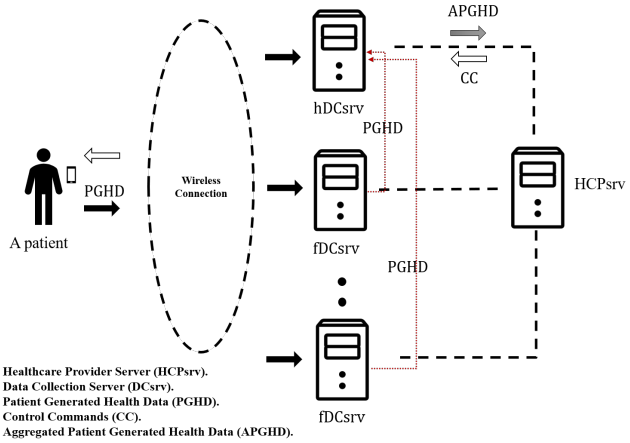

Figure 2: SPDC System Architecture

- Mobile device: A mobile device is a portable device (e.g. smartphone). It collects health data from wearable devices worn by the patient, classifies the data into normal or urgent data, and structures the data into a predefined format, called Patient-Generated Health Data (PGHD). I addition the data collected, the PGHD also contains a pseudonym ID and a flag indicating the type of data, normal (N) or urgent (U). For the sake of simplicity, in the remaining part of the paper, we use data to refer to data collected from patients
- Data collection servers DCsrvs: Data Collection servers (DCsrvs) are servers that are used to collect, store, aggregate and deliver the data collected to the healthcare provider server. If the data is an urgent one, it should be sent to the healthcare provider without further processing, (aggregation). Otherwise, it aggregates the data received in a given time interval and delivers them to their respective home servers. For a higher level of privacy preservation, different data collection servers may be owned by different service providers. Each patient is registered with one of these data collection servers but can use any of the servers to upload his/her data. The server registered by the patient is the patient's home data collection server, while other servers are the patient's foreign data collection servers. For different patients, their home data collection servers may be different. One data collection server may act as a home data collection server for one patient, but a foreign data collection server for another. The tasks performed by a data collection server are: (i) collects data from registered patients (as their home data collection server) and from unregistered patients (as their foreign data collection server), (ii) aggregates the normal data received from the registered patients before delivering the data to the healthcare provider and forwards the urgent data from the registered patients to the healthcare provider as soon as the data are received, (iii) forwards the normal data of the unregistered patients to their respective home data collection servers, (iv) generates authorization tokens for registered patients to allow them accessing foreign data collection servers (will be explained later). A patient can change his/her registration with a different data collection server anytime, thus the change of his/her home data collection server. A patient may also re-register with the same data collection server with a different pseudonym ID that is issued by the healthcare service provider. Frequent changes of home data collection servers or pseudonym IDs (i.e. re-registration) can enhance privacy protection level.
- The Healthcare Provider Server (HCPsrv) is a server owned by a healthcare service provider (e.g. a hospital). It collects, processes and stores patients' data. It also sends control commands (CC) to patients. The control commands specify frequencies at which certain data or data from certain patients should be collected, etc.

5 SPDC METHODS

On top of the SPDC architecture, there are two methods. The first is the pseudonym generation and linkage method. This method consists of a number of pseudonym generation and linkage algorithms. The second method is anonymous authentication method. This method includes two types of authentication namely, pseudonym certificate authentication and blind tokens authentication. Before we explain the methods in more details, we first specify the assumptions and notations used in these methods see Table 2. The assumptions are as follows.

(A1) The healthcare provider server (HCPsrv) is the certified authority which all the entities should be registered with.

(A2) Each entity (e) generates its own Elliptical key pair, a public and private key (EPK_e , EPR_e) and RSA key pair, a public and private key (PK_e , PR_e). Both EPK_e and PK_e are certified in digital certificates issued by the HCPsrv.

(A3) The patient signs his/her PHGD using a his elliptical curve private key, EPK_i. Then encrypts the PHGD and signature using a secret key know to both the patient and HCPsrv. This process is depicted in Figure 1.

(A4) The digital certificate for each data collection server is signed

by HCPsrv and uploaded onto an online repository.
(A5) The HCPsrv has already delegated his signature signing power by warrant to each data collection server. Then each data collection server generates its proxy signature key pair (PPK_{DC} , PPR_{DC}) based on the warrant.
(A6) Each data collection server generates a private key (K_p) and keep it secret. This key is used in blind signature generation.
(A7) Each data collection server generates their public parameters, (Rp , rp) and publishes them a long with the public proxy key (PPK_{DC}).

Table 2: SPDC Notations

Notation	Description
PGHD	Patient Generated Health Data
APGHD	Aggregated Patient Generated Health Data
CC	Control Commands
HCPsrv	HealthCare provider Server
DCsrvs	Data Collection Servers
hDCsrv	Home Data Collection Server
fDCsrvs	Foreign Data Collection Servers
PID_i	Patient Real ID for a patient i
MIP_i	Main Index Pseudonym for a patient i
HIP_i	Home Index Pseudonym for a patient i
FIP_i^f	Foreign Index Pseudonym for a patient i with a fDCsrv f
$HUP_{(i,r)}$	Home Uploading Pseudonyms for a patient i in uploading request r
$FUP_{(i,r)}^f$	Foreign Uploading Pseudonym for a patient i with a fDCsrv f in uploading request r
PK_e	The RSA Public key for an entity e
PR_e	The RSA Private Key for an entity e
EPK_e	The Elliptical Curve Public key for an entity e
EPR_e	The Elliptical Curve Private Key for an entity e
SK_{HCP}^1	HCPsrv First Secret Key
SK_{HCP}^2	HCPsrv Second Secret Key
SK_i	The Shared Key between a patient i and HCPsrv
HSK_i	The Home Shared Key for a patient i with hDCsrv
FSK_i^f	The Foreign Shared Key for a patient i with a fDCsrv f
PPK_e,PPR_e	Proxy Public and private Key for entity e
tPK_i^f ,tPR_i^f	The Temporal Public and private key generated by patient i with a fDCsrv f
Pesu-Cert	Pseudonym Certificate
Enc, Dec	Asymmetrical Encryption and Decryption
E, D	Symmetrical Encryption and Decryption
del	Delimiter
Pr	Priority Tag
RND	Random number generated
T	Time

5.1 Pseudonym Generation and Linkage Method

The SPDC architecture described above along with the multi-level hierarchical pseudonyms provide the patients' ID privacy preservation property. The pseudonyms for each patient are of five types structured at four hierarchical levels: (i) Main Index Pseudonym, (ii) Home Index Pseudonym, (iii) Foreign Index Pseudonyms and (iv) Home Uploading Pseudonyms and Foreign Uploading Pseudonyms. At the healthcare provider server, each patient is identified by his/her unique patient Index Pseudonym, called Main Index Pseudonym (MIP). This Main Index Pseudonym may be generated from the patient's real ID or another layer of pseudonym uniquely associated to the patient and only known to the healthcare provider server. As we explained previously, patient can use any of the servers to upload his/her data. For each server accessed by a patient, the patient is identified with a server-dependent pseudonym. This pseudonym is called the Home Index Pseudonym (HIP) if it is the patient's home server, or a Foreign Index Pseudonym (FIP) if it is a foreign server used by the patient. The HIP is generated based on MIP while, a FIP is generated based on the HIP. To further hide different uploadings by the same patient with the same server, each uploading is identified using a unique uploading pseudonym, another layer of pseudonyms called uploading pseudonyms, Home Uploading Pseudonyms (HUPs) and Foreign Uploading Pseudonyms (FUPs). These uploading pseudonyms are generated based on respective servers' index pseudonyms, i.e. Home Index Pseudonym or Foreign Index Pseudonyms. The structure of these different layers of pseudonyms is illustrated in Figure 3.

The linkage of two pseudonyms at different levels is a privilege. Such privileges are granted to entities on a need-to-know basis. The healthcare provider is allowed to link a patient's Home Index Pseudonym to his/her Main Index Pseudonym and then from the Main Index Pseudonym to the patient's real ID. A patient's home data collection server is allowed to link the patient's Foreign Index Pseudonyms and Home Uploading Pseudonyms to the patient's Home Index Pseudonym. A patient's foreign data collection server can link the patient's Foreign Uploading Pseudonyms to the patient's Foreign Index Pseudonyms. A patient has the same level of privileges in terms of linking his/her own pseudonyms.

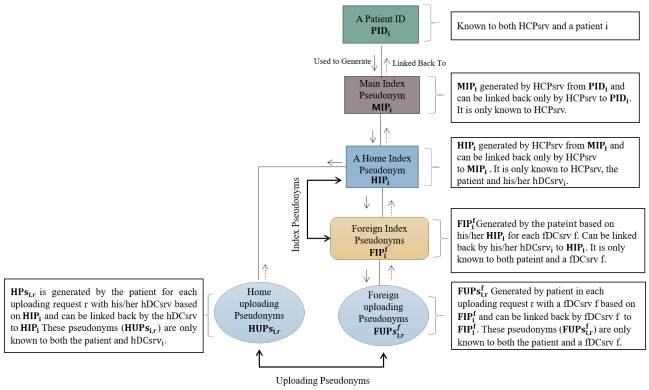

Figure 3: Pseudonym Types

5.2 Pseudonym Generation and Linkage Algorithms

There are four sets of algorithm for generating and linking the five types of pseudonym and they are as follows. Home Index Pseudonym Generation (HIP-Gen) and Linkage (HIP-LnK) algorithms, Foreign Index Pseudonyms (FIPs-Gen) and Linkage (FIPs-Lnk) algorithms, Home Uploading Pseudonyms Generation (HUPs-Gen) and Linkage (HUPs-LnK) algorithms and Foreign Uploading Pseudonyms Generation (FUPs-Gen) and Linkage (FUPs-LnK) algorithms.

5.2.1 Home-Index Pseudonym Generation and Linkage Algorithms. The Home-Index Pseudonym Generation (HIP-Gen) and Linkage (HIP-Lnk) algorithms are executed by HCPsrv by using two symmetric keys (SK^1_{HCP} , SK^2_{HCP}), that are only known to HCPsrv. The HIP-Gen algorithm uses Equations (1) and (2) and the HIP-Lnk algorithm uses Equations (3) and (4), where E/D are, respectively a symmetric encryption/decryption function, such as AES and other parameter values used in these equations (and other equations described in this section) are summarized in Table 2.

$$MIP_i = E(SK^1_{HCP}, PID_i||del||T_t) \tag{1}$$

$$HIP_i = E(SK^2_{HCP}, MIP_i||del||Rnd_t) \tag{2}$$

$$MIP_i||del||Rnd_t = D(SK^2_{HCP}, HIP_i) \tag{3}$$

$$PID_i||del||T_t = D(SK^1_{HCP}, MIP_i) \tag{4}$$

5.2.2 Foreign Index Pseudonyms Generation and Linkage Algorithms. The Foreign-Index Pseudonyms Generation (FIPs-Gen) and Linkage (FIPs-Lnk) algorithms are executed by a patient's mobile device and his/her home data collection server respectively by using home data collection key pair (i.e. a public and private key) (PK_{hDC_i} , PR_{hDC_i}). The FIPs-Gen algorithm uses Equation (5) and the FIPs-Lnk algorithm uses Equation (6), where Enc/Dec are, respectively asymmetric encryption/decryption function, such as RSA.

$$FIP^f_i = Enc(PK_{hDC_i}, HIP_i||del||T_f||Rnd_f) \tag{5}$$

$$HIP_i||del||T_f||Rnd_f = Dec(PR_{hDC_i}, FIP^f_i) \tag{6}$$

5.2.3 Home Uploading Pseudonyms Generation and Linkage Algorithms. The Home-Uploading Pseudonyms Generation (HUPs-Gen) and Linkage (HUPs -Lnk) algorithms are executed by a patient's mobile device and his/her home data collection server respectively by using a symmetric shared key (HSK_i). The HUPs-Gen algorithm uses Equation (7) and the HUPs-Lnk algorithm uses Equation (8).

$$HUPs_{i,r} = E(HSK_i, HIP_i||del||Pr||Rnd_r) \tag{7}$$

$$HIP_i||del||Pr||Rnd_r = D(HSK_i, HUPs_{i,r}) \tag{8}$$

5.2.4 Foreign Uploading Pseudonyms Generation and Linkage Algorithms. The Foreign-Uploading Pseudonyms Generation (FUPs-Gen) and Linkage (FUPs -Lnk) algorithms are executed by a patient's mobile device and a foreign data collection server respectively by using a symmetric shared key (FSK^f_i). The FUPs-Gen algorithm uses Equation (9) and the FUPs-Lnk algorithm uses Equation (10).

$$FUPs^f_{i,r} = E(FSK^f_i, FIP^f_i||del||Pr||Rnd_r) \tag{9}$$

$$FIP^f_i||del||Pr||Rnd_r = D(FSK^f_i, FUPs^f_{i,r}) \tag{10}$$

Next Section 6 shows how to use these pseudonyms in anonymous authentication.

6 ANONYMOUS AUTHENTICATION METHOD

This section describes two anonymous authentication methods, the pseudonym certificate based authentication method and the blind token based authentication method. The former is used to authenticate a patient to his/her home data collection server, while the latter is used to authenticate a patient to his/her foreign data collection server. The reason for using the two authentication methods, respectively, for accessing home and foreign data collection servers is to spread the overheads as introduced by the tasks of authentication (such as credential issuance and authentication token generation) across multiple entities, while still preserving patients' ID privacy. Both authentication methods are based on the Elliptical Curve Cryptosystem (ECC) [18].

6.1 Pseudonym Certificate based Authentication

The pseudonym certificate based authentication method involves the use of a pseudonym based authentication credential. Each patient is issued with such a credential by the healthcare provider server (HCPsrv). It consists of a certificate that binds the patient's ECC public key with his/her Home Index Pseudonym and the corresponding ECC private key only known to the patient. The structure of the pseudonym certificate (Pesu-Cert) is shown in Figure 4. The certificate contains the patient's home index pseudonym, his/er public key that is generated by the patient and other relevant information and it is signed by HCPsrv. During the authentication process, the patient submits the obtained certificate (i.e. Pesu-Cert) along with his signature one it to his/her home data collection server (hDCsrv). The hDCsrv performs the following verifications: i) verifies that the patient's signature is valid with the public key certified in the certificate, ii) verifies that the HCPsrv's signature in the certificate is valid with the HCPsrv's public key and iii) verifies that the certificate is not expired and has not been revoked.

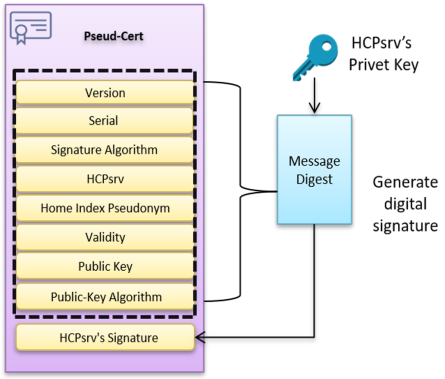

Figure 4: Pseudonym Certificate

6.2 Blind Token based Authentication

The blind tokens are used to authenticate the patient with foreign data collection servers, (fDCsrvs).The generation of a blind token is illustrated in Figure 5. As shown in the Figure, a patient, via the use of his/her mobile device, first constructs a data structure (i.e. token) with fields, the foreign index pseudonym (FIP), temporal public key (tPK), the ID of the patient's hDCsrv (hDCsrvID) and a validity period (Vpr) (the default setting is 24 hours). It then generates a hash value of these field values, blinds the hash value and sends it to the hDCsrv. The hDCsrv uses its proxy private key to sign the blinded hash value and returns it (i.e. the blind signature on the hash value) to the patient. The patient then unblinds the signature and appends the signature to the token in the field of the issuer's signature. Any fDCsrv who knows the proxy public key of the hDCsrv can verify the signature.

The construction of a blind token involved the use of a number of algorithms: Blind Token Generation (BlndTokGen) Algorithm, Blind Signature Generation (BlndSigGen) Algorithm, Blind Signature Driven (BlndSigDrv) Algorithm, Blind Token Construction (BlndTokCon) Algorithm and Blind Signature Verification (BlndSigVer) Algorithm. Both BlndSigGen and BlndSigDrv are adapted from [19]. Note that parameter values used in these algorithm are summarized in Table 2

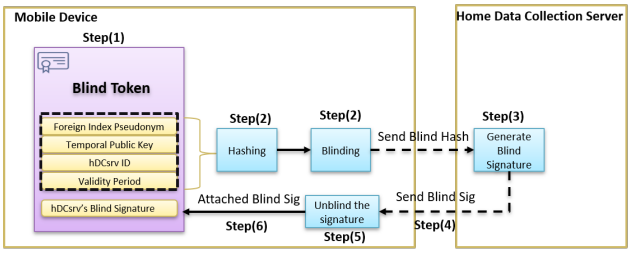

Figure 5: Blind Tokens Construction

6.2.1 Blind Token Generation (BlndTokGen) Algorithm. The algorithm consists of the following operations: (i) generating a temporal key pair (tPK_i^f , tPR_i^f), (ii) generating a foreign index pseudonym (FIP) using FIPs-Gen method,(iii) generating a token consisted of the temporal public key (tPK_i^f), foreign index pseudonym (FIP_i^f), validity period (Vpr) and the ID of the patient's hDCsrv (ID_{hDC}) i.e. token(tok) = { tPK_i^f , FIP_i^f , Vpr, ID_{hDC}}, (iv) generating R^* using Equation (11),where a,b and c are random blinding factors, (v) computing a hash value of the token using Equation (12), (vi) blinding the hash value e^* using Equation (13), (vii) requesting a blind signature on the hash value from hDCsrv.

$$R^* = aR_p + cP - bPPK_{hDC} \tag{11}$$

$$e^* = h(R^*||tok) \tag{12}$$

$$e = a^{-1}(e^* - b) \mod n \tag{13}$$

6.2.2 Blind Signature Generation (BlndSigGen) Algorithm. This algorithm is executed by the home data collection server (hDCsrv) of the patient. After receiving the blind token (e) from the patient, the hDCsrv generates the blind signature (S^*) on (e) using Equation (14) below. Then it sends the blind signature S^* to the patient.

$$S^* = ePPR_{hDC} + k_p \mod n \tag{14}$$

6.2.3 Blind Signature Driven (BlndSigDrv) Algorithm. This algorithm is executed by the mobile device of the patient. After receiving the blind signature (S^*) from hDCsrv, the patient, drives the unblind version of signature (S) using Equation (15) below.

$$S = S^*a + c \mod n \tag{15}$$

6.2.4 Blind Token Construction (BlndTokCon) Algorithm. This algorithm is executed by the mobile device of the patient. After driving the unblind version of signature (S) using Equation (16). The mobile device constructs the blind token by appending the signature (S) to the token. i.e. blndTok= { tPK_i^f , FIP_i^f , Vpr, ID_{hDC}, S}.

$$S = S^*a + c \mod n \tag{16}$$

6.2.5 Blind Signature Verification (BlndSigVer) Algorithm. This algorithm is executed by a foreign data collection server (fDCsrv). After receiving the the blind token ($BlndTok$) from the patient, the fDCsrv verifies the signature on the $BlndTok$ using Equation below. It extracts the four fields from the blind token (i.e. tok). Then it checks the validity using following Equation (17).

$$e^* = h(SP - e^*PPK_{hDC})||tok \tag{17}$$

The BlndTokGen, BlndSigDrv and BlndTokCon algoirthms are executed on the mobile device of the patient while the BlndSigGen algorithm is executed on the hDCsrv. The BlndSigVer is executed on mobile device and foreign data collection server.

7 SPDC REQUIREMENTS ANALYSIS

In this section we analyzed the SPDC against design requirements specified in Section 2.2.
Protocols for supporting various modes of data collections (F) will be presented in a future paper.

Preserving patient's ID privacy (P) is accomplished by using five types of pseudonym. We first discuss the number of attempts required to guess a patient's real ID based on his/her pseudonyms, and second discuss the probability to link multiple pseudonyms for the same patient. A patient's real ID is only known to the authorized entity (i.e. HCPsrv). For the patient's home and foreign data collection servers, they know the patient's home and foreign index pseudonyms respectively. To work out the patient's real ID from the Home Index Pseudonym (HIP), the unauthorized entity would need to reverse equation (2) to obtain the patient's MIP and then reverse equation (1). Using a brute force attack method, this involves guessing of two symmetric keys, which requires $2^{128} * 2^{128}$ attempts to reveal the real ID of the patient. To work out the patient's real ID from the Foreign Index Pseudonym (FIP), the unauthorized entity would need to reverse Equations (1), (2) and (5). This involves guessing of two symmetric keys and one asymmetric key, which requires $2^{128} * 2^{128} * 2^{2048}$ attempts. To work out the patient's real ID from a Home Uploading Pseudonym (HUP), the

unauthorized entity would need to reverse Equations (1), (2) and (7). Using a brute force attack method, this involves guessing of three symmetric keys, which requires $2^{128} * 2^{128} * 2^{128}$ attempts. To work out the patient's real ID from a Foreign Uploading Pseudonym (FUP), the unauthorized entity would need to reverse Equations (1), (2), (5) and (9). Using a brute force attack method, this involves guessing of three symmetric keys and one asymmetrical key which requires $2^{128} * 2^{128} * 2^{2048} * 2^{128}$ attempts. Table 4 illustrates the symmetrical key size and the time required to crack it. In addition, the hierarchical pseudonyms are protected against forgery and replay attacks, as their generations involve the use of time and random numbers. Moreover, SPDC tries to break any linkage of frequencies of uploading (uploading pattern) performed by the same patient, thus further strengthening the protection of the patient's ID privacy. Each patient can select a set of foreign data collection servers along with a home data collection server to upload their data. The selection of the foreign data collection servers can be done randomly both in terms of servers' IDs and the number of the set. In addition, the accesses to these different servers are based on different pseudonyms and different uploadings with a given server are also identified by using different pseudonyms. Furthermore, the registration with a home data collection server can also change at different time intervals. Later on in this paper, we discuss the probability of linking multiple pseudonyms for the same patient using the entropy-based method.

Support for mutual authentication (S1) is accomplished by using pseudonym certificates, blind authentication tokens and digital signatures. Mutual authentication between a patient and a data collection server is achieved through digital signatures on a fresh nonce contributed by the respective signature verifier. For the patients, the signature verification keys are certified in the form of pseudonym certificates (for accessing hDCsrvs) or short-term blind tokens (for accessing fDCsrvs), whereas the corresponding signature signing keys are only known to their respective patients. Provided that the signature signing key is secure and the nonce on which the signature is signed is random and fresh, it is hard for another entity to impersonate the signer. Similarly, for the data collection servers, the signature verification keys are certified in a form of the certificate. Support for data authenticity and confidentiality (S2 and S3). This achieved through sign then encrypt process. The patient generates a digital signature by signing his/her PGHD and a fresh nonce using ECC private key that its corresponding verification key (i.e. ECC public key) is certified in a form of a pseudonym certificate. The digital signature, PGHD and the fresh nonce are then encrypted using a shared key known to both the patient and the HCPsrv. Unauthorized entity will not be able to tamper with the patient's data or launch a reply attack.

Degree of Anonymity. The degree of anonymity provided by SPDC is measured by using an entropy-based method [20]. The method shows how indistinguishable users of the system are from the attacker (i.e. if the attacker can determine a particular user to be the generator of a message by any means, then this means that the degree of anonymity provided by this system is low).

To calculate the degree of anonymity provided by SPDC, we assume that X be the discrete random variable which represents an uploading pseudonym. This uploading pseudonym can be correctly linked to a certain patient i. From a set of N possible patients that are involved $p_i = Pr(X = i)$. The current entropy H(X) of the corresponding uploading pseudonym can be calculated as follows.

$$H(X) = -\sum_{i=0}^{n} P_i \log(p_i) \tag{18}$$

The maximum entropy of the system H(M) can be calculated as

$$H(M) = \log N \tag{19}$$

The degree of anonymity (d) provided by SPDC can be calculated as.

$$d = H(X)/H(M), 0 \leq d \leq 1. \tag{20}$$

Suppose that number of patients who is using our system is 100 patients (i.e. N=100). Each patient generates a number of uploading pseudonyms . The attacker will not be able to distinguish one particular patient as being the owner of these uploading pseudonyms. What the attacker can do is the following. The attacker can divide the patients into two groups (G_1, G_2). Then assigns for each group a probability as follows.

$$G_1 = p/40, 0 \leq i \leq 40 \quad G_2 = 1 - p/60 \quad 41 \leq i \leq 60 \tag{21}$$

In the above equation the attacker assigns equal probability for each patient in the same group. This means that the attacker is unable to distinguish one patient of being the potential owner of a set of uploading pseudonyms. In Figure 6 we can see the degree of anonymity (d) is proportional to the number of the patients. We can see that when the number of patient is (N=100), the maximum degree of anonymity (d = 1) is achieved for the probability distribution (p = 0.41). The degree of anonymity is equal to 0.8 when one of the groups is assigned the probability p=0.93 and a number of patients is (N=10000). However, the degree of anonymity does not drop to zero as the attacker sees all patients as the potential owner of the uploading pseudonyms.

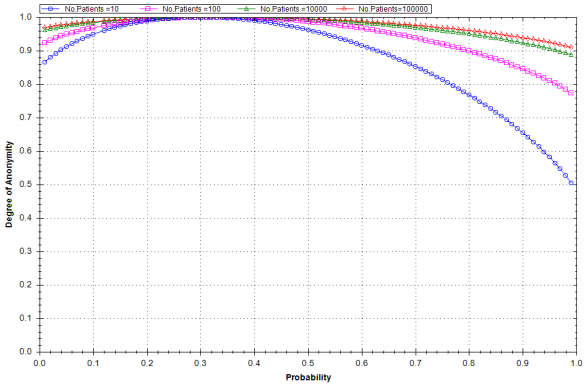

Figure 6: Degree of Anonymity

8 CONCLUSION

This paper has described the system architecture and its associated methods of a novel framework, the SPDC framework, designed to support secure and privacy-preserving data collection from mobile patients. Different from related solutions in literature, the SPDC

Table 3: AES Key Size and Time to Crack

Key Size	Time to Crack
56-bit	399 seconds
128-bit	$1.02*10^{18} years$
192-bit	$1.872*10^{37} years$
256-bit	$3.31*10^{56} years$

architecture support the use of multiple data collection servers in a structured manner such that, on per patient's basis, servers are classified into home and foreign data collection servers. This structured use of multiple servers approach, combined with the use of hierarchical pseudonyms, data aggregation, the principle of the separation of duties and distributed load sharing, brings us a number of merits. Firstly, in addition to supporting patients' anonymity (through the use of multi-level pseudonyms) while uploading their data, their server access patterns can also be hidden by allowing the patients to select different servers and different number of servers to upload their data. This latter facility can help to reduce the risk of compromising a patient's real ID through observing his/er server access patterns and then collating such patterns with information obtained elsewhere. The approach can also help to improve the scalability of the system, as SPDC functions can now be spread over multiple entities, taking some processing load off the healthcare provider server, reducing the risk of making it a performance bottleneck in the system. Data collected from patients are classified into two groups, normal and urgent data. Data aggregation is applied to normal data at every point of data collection, reducing bandwidth costs in delivering them to the final destination server, the healthcare provider server. The paper has also described the methods required to support core SPDC functions, namely methods for the generation and linkage of hierarchical pseudonyms and two anonymous authentication methods, one for the authentication of patients with their home data collection servers and the other the authentication of patients with their foreign home data collection servers. Future work includes quantitative evaluations of the architecture and the methods used and protocol designs to facilitate multi-mode data collections using the architecture.

REFERENCES

[1] A. Al-Fuqaha, M. Guizani, M. Mohammadi, M. Aledhari, and M. Ayyash, "Internet of things: A survey on enabling technologies, protocols, and applications," *IEEE Communications Surveys & Tutorials*, vol. 17, no. 4, pp. 2347–2376, 2015.

[2] S. R. Islam, D. Kwak, M. H. Kabir, M. Hossain, and K.-S. Kwak, "The internet of things for health care: a comprehensive survey," *IEEE Access*, vol. 3, pp. 678–708, 2015.

[3] G. Paliwal and A. W. Kiwelekar, "A comparison of mobile patient monitoring systems," in *International Conference on Health Information Science*, pp. 198–209, Springer, 2013.

[4] P. Pawar, V. Jones, B.-J. F. Van Beijnum, and H. Hermens, "A framework for the comparison of mobile patient monitoring systems," *Journal of biomedical informatics*, vol. 45, no. 3, pp. 544–556, 2012.

[5] M. reporter, "The nhs is about to take an âĂŸimportantâĂŹ step into the cloud, says microsoft," Jan. 2018.

[6] P. Kakria, N. Tripathi, and P. Kitipawang, "A real-time health monitoring system for remote cardiac patients using smartphone and wearable sensors," *International journal of telemedicine and applications*, vol. 2015, p. 8, 2015.

[7] M. Quwaider and Y. Jararweh, "Cloudlet-based efficient data collection in wireless body area networks," *Simulation Modelling Practice and Theory*, vol. 50, pp. 57–71, 2015.

[8] Q. Shen, X. Liang, X. S. Shen, X. Lin, and H. Y. Luo, "Exploiting geo-distributed clouds for a e-health monitoring system with minimum service delay and privacy preservation," *IEEE journal of biomedical and health informatics*, vol. 18, no. 2, pp. 430–439, 2014.

[9] M. Chen, Y. Qian, J. Chen, K. Hwang, S. Mao, and L. Hu, "Privacy protection and intrusion avoidance for cloudlet-based medical data sharing," *IEEE transactions on Cloud computing*, 2016.

[10] A. Lounis, A. Hadjidj, A. Bouabdallah, and Y. Challal, "Secure and scalable cloud-based architecture for e-health wireless sensor networks," in *2012 21st International Conference on Computer Communications and Networks (ICCCN)*, pp. 1–7, IEEE, 2012.

[11] E. Marin, M. A. Mustafa, D. Singelée, and B. Preneel, "A privacy-preserving remote healthcare system offering end-to-end security," in *International Conference on Ad-Hoc Networks and Wireless*, pp. 237–250, Springer, 2016.

[12] S. R. Moosavi, T. N. Gia, E. Nigussie, A. M. Rahmani, S. Virtanen, H. Tenhunen, and J. Isoaho, "End-to-end security scheme for mobility enabled healthcare internet of things," *Future Generation Computer Systems*, vol. 64, pp. 108–124, 2016.

[13] M. Layouni, K. Verslype, M. T. Sandıkkaya, B. De Decker, and H. Vangheluwe, "Privacy-preserving telemonitoring for ehealth," in *IFIP Annual Conference on Data and Applications Security and Privacy*, pp. 95–110, Springer, 2009.

[14] K. Mtonga, H. Yang, E.-J. Yoon, and H. Kim, "Identity-based privacy preservation framework over u-healthcare system," in *Multimedia and Ubiquitous Engineering*, pp. 203–210, Springer, 2013.

[15] X. Lin, R. Lu, X. Shen, Y. Nemoto, and N. Kato, "Sage: a strong privacy-preserving scheme against global eavesdropping for ehealth systems," *IEEE Journal on Selected Areas in Communications*, vol. 27, no. 4, pp. 365–378, 2009.

[16] X. Liang, R. Lu, L. Chen, X. Lin, and X. Shen, "Pec: A privacy-preserving emergency call scheme for mobile healthcare social networks," *Journal of Communications and Networks*, vol. 13, no. 2, pp. 102–112, 2011.

[17] P. Gope and T. Hwang, "Bsn-care: A secure iot-based modern healthcare system using body sensor network," *IEEE Sensors Journal*, vol. 16, no. 5, pp. 1368–1376, 2016.

[18] C. Paar and J. Pelzl, *Understanding cryptography: a textbook for students and practitioners*. Springer Science & Business Media, 2009.

[19] C.-H. Wang and M.-Z. Liao, "Security analysis and enhanced construction on ecdlp-based proxy blind signature scheme," *International Journal of e-Education, e-Business, e-Management and e-Learning*, vol. 4, no. 1, p. 47, 2014.

[20] Y. Deng, J. Pang, and P. Wu, "Measuring anonymity with relative entropy," in *International Workshop on Formal Aspects in Security and Trust*, pp. 65–79, Springer, 2006.