INVITED PAPER *Special Section on Security, Privacy and Anonymity of Internet of Things*

Remote Data Integrity Checking and Sharing in Cloud-Based Health Internet of Things

Huaqun WANG[†,††a)], **Keqiu LI**[†††b)], ***Nonmembers***, **Kaoru OTA**[††††c)], ***Member***, ***and*** **Jian SHEN**[†d)], ***Nonmember***

SUMMARY In the health IoT (Internet of Things), the specialized sensor devices can be used to monitor remote health and notify the emergency information, e.g., blood pressure, heart rate, etc. These data can help the doctors to rescue the patients. In cloud-based health IoT, patients' medical/health data is managed by the cloud service providers. Secure storage and privacy preservation are indispensable for the outsourced medical/health data in cloud computing. In this paper, we study the integrity checking and sharing of outsourced private medical/health records for critical patients in public clouds (ICS). The patient can check his own medical/health data integrity and retrieve them. When a patient is in coma, some authorized entities and hospital can cooperate to share the patient's necessary medical/health data in order to rescue the patient. The paper studies the system model, security model and concrete scheme for ICS in public clouds. Based on the bilinear pairing technique, we design an efficient ICS protocol. Through security analysis and performance analysis, the proposed protocol is provably secure and efficient.

key words: *remote data integrity checking, public cloud, data sharing*

1. Introduction

IoT can be used in the healthcare applications. It plays a significant role from managing chronic diseases at one end of the spectrum to preventing disease at the other, such as clinical care and remote monitoring. In the life of every people, the generated medical data and the health data is very large. In the medical research, the researchers will process massive medical health data. Massive data processing and information security risk call for the new computation model as an alternative to conventional computing. As a new computation model, cloud computing has become a reality along with the development of network and computer technology. Cloud computing provides a flexible, dynamic, resilient and cost effective infrastructure for the business environments. For the patients and hospital, the remote medical/health data integrity checking and sharing can be performed by the patients and hospital. Since the medical/health data is outside the control of patients and hospital, cloud service providers are more responsible for the security of application services, especially in public clouds. Based on public cloud server's benefits and security risks, the paper focuses on privacy-preserving remote medical/health data integrity checking and sharing for critical patients in public clouds.

Throughout the paper, "privacy-preserving ICS" is simplified as "ICS", "public cloud server" is simplified as "PCS".

1.1 Motivation

Along with the development of IoT, it is widely used in the field of healthcare. By using the IoT, massive health data is generated. On the other hand, when the patient goes to the hospital, the medical data is also generated. These generated medical/health data can be used in order to rescue the patients. In the cloud-based health IoT, the medical/health data is stored in public clouds. In public clouds, the hospitals and patients access PCS *via* Internet. To protect the hospital's benefits, it is important to prevent unauthorized entities to check these data integrity. If the malicious competitors can check these data integrity, they can evaluate these data size. Then, the competitors can evaluate the hospital's daily business volume. Then, the competitors can take measures to prevail the hospital. In order to protect these data, the remote data integrity checking can only be performed by the hospital besides of the patient. Usually, in order to protect the patients' privacy, only the patient can retrieve his own medical/health data. Unfortunately, the critical patients may be in a coma before reaching the hospital. In this case, the hospital and the patient's relatives should be able to cooperate to share the patient's medical/health data in order to rescue him. This real social requirement motivates us to study privacy-preserving remote medical/health data integrity checking and sharing for critical patients in public clouds.

1.2 Related Work

The rapidly developed IoT has been widely applied in the medical/health field [1], [2]. Based on the special properties of medical/health data, when IoT is used in the medical/health field, some security risks emerge [3], [4]. In 2015, Wu *et al.* proposed employment of the regenerating codes

Manuscript received November 24, 2015.
Manuscript revised April 12, 2016.
Manuscript publicized May 31, 2016.

[†] The authors are with the School of Computer & Software, Nanjing University of Information Science & Technology, China.

[††] The author is with the School of Computer, Nanjing University of Posts and Telecommunications, China.

[†††] The author is with the School of Computer Science and Engineering, Dalian University of Technology, China.

[††††] The author is with the Department of Information and Electronic Engineering, Muroran Institute of Technology, Muroran-shi, 050–8585 Japan.

a) E-mail: wanghuaqun@aliyun.com
b) E-mail: likeqiu@gmail.com
c) E-mail: ota@csse.muroran-it.ac.jp
d) E-mail: s_shenjian@126.com

DOI: 10.1587/transinf.2015INI0001

and symmetric-key encryption with a Blom based key management [5]. Their scheme can repair the lost fragment and protect data secrecy. When the medical/health data is stored on the PCS, the remote data integrity checking and sharing become an urgent security problem which needs to be solved. In 2014, Wang *et al.* proposed the fair remote retrieval of outsourced private medical records in electronic health networks [6]. Lu *et al.* proposed privacy preserving opportunistic computing framework for mobile-health care emergency [7]. In 2016, He *et al.* introduced a novel network security architecture for cloud computing considering characteristics of cloud computing [8]. Their scheme can protect external and internal traffics in cloud computing. It can also attain flexible scalability with respect to virtual middle box load and achieve fault-tolerant among virtual middle box failure.

Since these privacy-aware medical/health data is stored on the untrusted third party PCS, it is important to keep these data uncorrupted and privacy-preserving. As an efficient remote data integrity checking model, the concept of provable data integrity (PDP) was proposed by G. Ateniese *et al.* in 2007 [9]. They also designed two concrete statically secure PDP schemes. Since PDP is a very efficient remote data integrity checking model, many researchers proposed a variety of PDP security models and concrete schemes [10]–[14].

Privacy-preservation is an important security issue in cloud computing. In 2013, Wang *et al.* proposed the privacy-preserving public auditing in cloud computing [15]. Nabeel *et al.* proposed privacy preserving policy based content sharing in public clouds [16]. Guo *et al.* proposed the variable threshold-value authentication architecture for wireless mesh networks [17]. In the medical/health data cloud storage, privacy preservation is especially important. Only the patient and the authorized entities can get the patient's medical/health data. PDP protocols are classified into two categories: private PDP and public PDP. Some private information is necessary to perform private PDP. The private information is not needed for public PDP.

1.3 Our Contribution and Organization

In this paper, we propose the novel concept of ICS. Then, we give the formal system model and security model of ICS. By using the bilinear pairings, an efficient ICS protocol is designed. Through security analysis and performance analysis, our ICS protocol is provably secure and efficient.

The rest of the paper is organized as follows. Section 2 introduces the preliminaries and Sect. 3 describes our pairing-based ICS protocol and analyzes its security. Section 4 gives the performance analysis of our ICS protocol. Finally, Sect. 5 concludes this paper.

2. Preliminaries

ICS system model and security model are proposed in this section. After that, bilinear pairings and some corresponding difficult problems are also described below.

2.1 System Model and Security Model

ICS system consists of four different network entities: *Patient, Hospital, AuthSet, PCS*. They can be identified below.

1. *Patient*, whose medical/health data will be uploaded to PCS for maintenance and computation, is individual human being;
2. *Hospital*, which diagnoses the patients and generates the medical/health data for the patients, can be physicians or medical institutions;
3. *AuthSet*, which is the patient's authorized entity set, can cooperate with the hospital to share the patient's medical/health data;
4. *PCS*, which is managed by cloud service provider, has significant storage space and computation resource to maintain the patients' data.

Definition 1 (ICS). *A ICS protocol is a collection of seven polynomial time algorithms (SetUp, EncTagGen, CheckTagSign, GenProof, GenRetr, CheckProof, Retrieval) among PCS, Hospital, Patient, and AuthSet such that:*

1. *SetUp*$(1^k) \rightarrow (params, sk, pk)$ *is a probabilistic system parameters and key generation algorithm to setup the protocol where k is a security parameter,* sk *is the secret key and* pk *is the public key.* params *is the public system parameters.* ID_j *gets its private/public key pair* (x_j, X_j) *and a symmetric encryption key* sk_j*. The hospital's private/public key pair are* (y, Y)*. PCS's private/public key pair are* (z, Z)*. The patient* ID_j *prepares the warrant* ω_j *and the corresponding certificate Sign*(ω_j)*.* ID_j *creates* sk_j*'s secret shares and distributes the shares among AuthSet*$_j$.
2. *EncTagGen*$(x_j, sk_j, Y, Z, M_j) \rightarrow \{T_{i,j}\}$ *is run by the patient* ID_j *to generate the verification metadata, where* M_j *is* ID_j*'s medical/health data and* $T_{i,j}$ *is* ID_j*'s i-th block's tag.*
3. *CheckTagSign*$(F_{i,j},\ T_{i,j},\ z,\ X_j,\ Y,\ Z,\ \omega_j,\ Sign(\omega_j)) \rightarrow \{``success", ``failure"\}$ *is run by PCS to check whether the medical/health block-tag pair* $(F_{i,j}, T_{i,j})$ *and the warrant-signature pair* $(\omega_j,\ Sign(\omega_j))$ *are valid or not.*
4. *GenProof*$(F, chal_p, \Sigma) \rightarrow V_p$ *is run by PCS to generate the proof of integrity.* F *is the stored file and* Σ *is the stored tags.* $chal_p$ *is the challenge from the verifier.*
5. *GenRetr*$(F, chal_r, \Sigma) \rightarrow V_r$ *is run by PCS to generate the retrieval response.*
6. *CheckProof*$(x_j \text{ or } y, X_j, Y, Z, chal_p, V_p) \rightarrow \{``success", ``failure"\}$ *is run by the patient* ID_j *or the hospital to check the data integrity.*
7. *Retrieval*$((x_j, sk_j) \text{ or } (y, sk'_j s \text{ valid share set}), X_j, Y, Z, chal_r, V_r) \rightarrow M_j$ *is run by the patient* ID_j *or the cooperation of hospital and AuthSet*$_j$ *to retrieve* ID_j*'s remote medical/health data* M_j*.*

The following definitions 2, 3, 4 define the security against the malicious PCS forgery, restrictive remote medical/health data integrity checking and restrictive retrievability.

Definition 2 (Integrity Against Malicious PCS). *ICS protocol satisfies the integrity if any PPT (probabilistic polynomial time) adversary $\mathcal{A}$ (*i.e., *malicious PCS) can win the ICS game only with negligible probability. ICS game between the challenger C and the adversary $\mathcal{A}$ is described below:*

1. *Setup: In this phase, the system parameters* params *are created. Let the patient set be $\mathcal{P}$. The patient ID_j's private/public key pair (x_j, X_j), the hospital's private/public key pair (y, Y) and PCS's private/public key pair (z, Z) are created where $ID_j \in \mathcal{P}$. Let ID_j's symmetric encryption/decryption key be* sk_j. *The private keys x_j, y are kept secret. The parameters (z, X_j, Y, Z, params, $ID_j \in \mathcal{P}$) are sent to $\mathcal{A}$.*
2. *First-Phase Queries: $\mathcal{A}$ adaptively queries C below.*
   - *Hash query. Input the hash queries adaptively, C responds the corresponding hash values to $\mathcal{A}$.*
   - *Tag query. Input the medical/health block $F_{i,j}$ for the patient ID_j, C calculates the tag $T_{i,j} \leftarrow TagGen(x_j, F_{i,j})$ and sends it to $\mathcal{A}$. Without loss of generality, let $\{(F_{i,j}, T_{i,j})\}$ be the queried block-tag pair set and $\mathbb{I}_1 = \{(i, j)\}$ in First-Phase Queries.*
3. *Challenge: C generates a data integrity challenge $chal_p$ which defines the challenged block-tag pair index collection $\mathbb{I}_c = \{(i_1, j_1), (i_2, j_2), \cdots, (i_c, j_c)\}$, where $\mathbb{I}_c \nsubseteq \mathbb{I}_1$ and c is a positive integer. C is queried to provide the proof of integrity checking for the medical/health blocks $F_{i_1,j_1}, \cdots, F_{i_c,j_c}$.*
4. *Second-Phase Queries: Similar to First-Phase Queries. Let the queried medical/health block-tag pair set be $\{(F_{i,j}, T_{i,j})\}$ and $\mathbb{I}_2 = \{(i, j)\}$ in Second-Phase Queries. The restriction is that $\mathbb{I}_c \nsubseteq (\mathbb{I}_1 \cup \mathbb{I}_2)$.*
5. *Forge: Finally, $\mathcal{A}$ forges a data integrity proof V_p for the medical/health blocks indicated by $chal_p$ and returns V_p to C.*

In the above ICS game, we say that $\mathcal{A}$ wins if

$$\Pr\left[\begin{array}{l} CheckProof(y, X_j, Y, Z, chal_p, V_p, \\ ID_j \in \mathcal{P}) \rightarrow \text{“success”} \end{array}\right] \geq \frac{1}{p(k)}$$

where $p(k)$ is a polynomial of the security parameter k.

Definition 3 (Restrictive Integrity Checking). *In the remote medical/health data integrity checking, only the following restrictive entities have the ability to perform the data integrity checking protocol:*

1. *The hospital can perform the remote medical/health data integrity checking for all the patients.*
2. *The individual patient can perform the integrity checking only for his own remote medical/health data.*
3. *Except the hospital, patient and PCS, the other entity cannot perform the remote medical/health data integrity checking.*

Definition 4 (Restrictive Retrievability). *In the remote medical/health data retrievability, the patient ID_j creates the retrievability control set $\mathcal{R} = \{R_{1,j}, R_{2,j}, \cdots, R_{\hat{n}_j,j}\}$. It satisfies the following requirements:*

1. *The patient ID_j can retrieve his own remote medical/health data by himself,* i.e., $\{ID_j\} \in \mathcal{R}$.
2. *For the other retrievability control set $R_{i,j} \in \mathcal{R}$, the entities in $R_{i,j}$ can cooperate to retrieve ID_j's remote medical/health data under the help of the hospital.*
3. *For any entity set R_A, if $R_{i,j} \nsubseteq R_A$ for every $1 \leq i \leq \hat{n}_j$, the entities in S_A cannot retrieve ID_j's remote medical/health data even if they collude.*

A secure ICS protocol also needs to guarantee that after validating the PCS-generated proof, the verifier can also be convinced that all of his outsourced data has been kept intact with a high probability. The following security definition gives the security property.

Definition 5 ((ρ, δ) Security). *An ICS protocol is (ρ, δ)-secure if PCS corrupted ρ fraction of the whole medical/health blocks, the probability that the corrupted blocks are detected is at least δ.*

2.2 Bilinear Pairings and Difficult Problems

Let $\mathcal{G}_1$ and $\mathcal{G}_2$ be two cyclic multiplicative groups with the same prime order q. Let $\hat{e} : \mathcal{G}_1 \times \mathcal{G}_1 \rightarrow \mathcal{G}_2$ be a bilinear map. $\hat{e}$ can be constructed by the modified Weil or Tate pairings on elliptic curves [18], [19]. The group with such a map $\hat{e}$ is called a bilinear group, on which the Computational Diffie-Hellman (CDH) problem is assumed hard while the Decisional Diffie-Hellman (DDH) problem is easy [20].

Definition 6 (Gap Diffie-Hellman (GDH) Group). *Let g is the generator of $\mathcal{G}_1$. Given $g, g^a, g^b, g^c \in \mathcal{G}_1$ while $a, b, c \in \mathcal{Z}_q^*$ are unknown, it is recognized that there exists an efficient algorithm to determine whether $ab = c \bmod q$ holds by verifying $\hat{e}(g^a, g^b) = \hat{e}(g, g^c)$ in polynomial time (DDH problem), while there exist no efficient algorithms to compute $g^{ab} \in \mathcal{G}_1$ with non-negligible probability within polynomial time (CDH problem). An algorithm $\mathcal{A}$ is said to (t, ϵ)-break the CDH problem on $\mathcal{G}_1$ if $\mathcal{A}$ runs in time at most t, and the following CDH advantage is at least ϵ.*

$$Adv_{\mathcal{A}}^{CDH} = \Pr[\mathcal{A}(g, g^a, g^b) \rightarrow g^{ab} : \forall a, b \in \mathcal{Z}_q^*] \geq \epsilon$$

The probability is taken over the choice of a, b and $\mathcal{A}$'s coin tosses. A group $\mathcal{G}_1$ is a (t, ϵ)-GDH group if the DDH problem on $\mathcal{G}_1$ is efficiently computable and no algorithm (t, ϵ)-breaks the CDH problem on $\mathcal{G}_1$.

Definition 7 (Bilinear Diffie-Hellman Problem (BDHP) assumption). *Given (g, g^a, g^b, g^c) for unknown $a, b, c \in \mathcal{Z}_q^*$, it is difficult to compute $W = \hat{e}(g, g)^{abc} \in \mathcal{G}_2$.*

3. Our Proposed Pairing-Based ICS Protocol

3.1 ICS Protocol Construction

Our ICS protocol consists of seven phases: (*SetUp, EncTagGen, CheckTagSign, GenProof, GenRetr, CheckProof, Retrieval*) among PCS, *Hospital*, *Patient*, and *AuthSet*. Suppose there are n_p patients whose set is denoted as $\mathcal{P}$, *i.e.*, $\mathcal{P} = \{ID_1, ID_2, \cdots, ID_{n_p}\}$. Suppose the patient ID_j will upload n_j block-tag pairs. Denote the corresponding block-patient index set as $\mathcal{B}\times\mathcal{P} = \{(i_j, j), 1 \le j \le n_p, 1 \le i_j \le n_j\}$. Let n denote the whole block number, *i.e.*, $n = \sum_{j=1}^{n_p} n_j$. Let f and Ω be two pseudo-random functions, and let π be a pseudo-random permutation. Let H, h be two cryptographic hash functions. They are described below.

$$\begin{aligned}
&f : \mathcal{Z}_q^* \times \{1, 2, \cdots, n\} \to \mathcal{Z}_q^*,\\
&\Omega : \mathcal{Z}_q^* \times (\mathcal{B}\times\mathcal{P}) \to \mathcal{Z}_q^*\\
&\pi : \mathcal{Z}_q^* \times \{1, 2, \cdots, n\} \to \mathcal{B}\times\mathcal{P}\\
&H : \mathcal{G}_2 \times \{0, 1\}^* \to \mathcal{Z}_q^*, \quad h : \mathcal{Z}_q^* \to \mathcal{G}_1^*
\end{aligned}$$

Let g be a generator of $\mathcal{G}_1$. Without loss of generality, we only consider the patient ID_j in the concrete scheme construction. Denote ID_j's medical/health data as M_j. The patient ID_j picks a random $x_j \in \mathcal{Z}_q^*$ as his private key and computes $X_j = g^{x_j}$ as his public key. On the other hand, ID_j picks a random $sk_j \in \mathcal{Z}_q^*$ as his symmetric encryption/decryption key. The hospital picks a random number $y \in \mathcal{Z}_q^*$ as its private key and computes $Y = g^y$ as its public key. PCS picks a random number $z \in \mathcal{Z}_q^*$ as its private key and computes $Z = g^z$ as its public key. The phases of our proposed ICS protocol are described below.

SetUp: The patient ID_j delegates his remote medical/health data retrieval capability to the authorized entity set $AuthSet_j$. Suppose that n_{s_j} entities share ID_j's symmetric key and th_j entities can retrieve ID_j's remote medical/health data by cooperating with the hospital. Let the authorized entity set be $AuthSet_j = \{D_{j,1}, D_{j,2}, \cdots, D_{j,s_j}\}$ and $D_{j,i} \in \mathcal{Z}_q^*$ (it can be realized by using the hash function $\mathbb{H} : \{0, 1\}^* \to \mathcal{Z}_q^*$). Let $(Sign, Verify)$ be secure signature/verification algorithm pairs. In order to restrict the authorized entities' behaviors, for every entity $D_{j,i}$, ID_j creates the warrant ω_j and $Sign(\omega_j)$ by using his own private key x_j. The warrant ω_j describes the rules which must be obeyed by $D_{j,i}$. For the symmetric encryption key sk_j, ID_j generates the corresponding shares below.

1. Pick the random $a_{j,1}, a_{j,2}, \cdots, a_{j,th_j-1} \in \mathcal{Z}_q^*$ and get the following polynomial with the order $th_j - 1$:

$$\mathcal{F}_j(x) = sk_j + \sum_{i=1}^{th_j-1} a_{j,i}x^i$$

2. Compute the share $ss_{j,i} = \mathcal{F}_j(D_{j,i})$ for every entity $D_{j,i}$. Then, ID_j sends $(ss_{j,i}, \omega_j, Sign(\omega_j))$ to $D_{j,i}$.
3. Every entity $D_{j,i}$ verifies whether the warrant-signature pair $(\omega_j, Sign(\omega_j))$ is valid. If it is valid, $D_{j,i}$ keeps the warrant-signature pair $(\omega_j, Sign(\omega_j))$ and the corresponding secret share $ss_{j,i}$ of the symmetric encryption/decryption key sk_j of the patient ID_j.

Finally, ID_j picks a public random element $u_j \in \mathcal{G}_1^*$. The final system parameters are

$$params = \{\mathcal{G}_1, \mathcal{G}_2, \hat{e}, f, \Omega, \pi, H, h, X_j, Y, Z, u_j, q, ID_j \in \mathcal{P}\}$$

EncTagGen(x_j, sk_j, M_j, Ω, Y, Z): After finishing the medical/health advices, the patient ID_j gets his own medical/health data M_j. Taking use of the symmetric encryption algorithm, ID_j gets the ciphertext $F_j = E_{sk_j}(M_j)$. Then, F_j is divided into n_j blocks, *i.e.*, $F_j = (F_{1,j}, F_{2,j}, \cdots, F_{n_j,j})$. For the block $F_{i,j}$, the patient ID_j computes $t_j = H(\hat{e}(Y, Z)^{x_j}, \omega_j)$, $W_{i,j} = \Omega_{t_j}(i, j)$ and Compute $T_{i,j} = (h(W_{i,j})u_j^{F_{i,j}})^{x_j}$. Then, it outputs the block-tag pair $(F_{i,j}, T_{i,j})$.

When the above procedures are performed n_j times, all the block-tag pairs are created. At last, the patient ID_j uploads his block-tag pairs collection $\{(F_{i,j}, T_{i,j}), 1 \le i \le n_j\}$ and the warrant-signature pairs $(\omega_j, Sign(\omega_j))$ to PCS. PCS stores the block-tag pairs and the warrant-signature pairs $(\omega_j, Sign(\omega_j))$. The patient deletes these block-tag pairs $\{(F_{i,j}, T_{i,j}), 1 \le i \le n_j\}$ from its local storage.

CheckTagSign($\{z, X_j, Y, (\omega_j, Sign(\omega_j)), (F_{i,j}, T_{i,j}), 1 \le i \le n_j\}$): Upon receiving $(\omega_j, Sign(\omega_j))$, PCS verifies its validity by using the corresponding verification algorithm *Verify*. If it is invalid, output "failure". Otherwise, for every $1 \le i \le n_j$, PCS computes $\hat{t}_j = H(\hat{e}(X_j, Y)^z, \omega_j)$ and $\hat{W}_{i,j} = \Omega_{\hat{t}_j}(i, j)$. Then, it verifies whether the following formula holds: $\hat{e}(T_{i,j}, g) \overset{?}{=} \hat{e}(h(\hat{W}_{i,j})u_j^{F_{i,j}}, X_j)$. If it holds, then PCS accepts it. Otherwise, PCS rejects it and responds "failure".

GenProof($F, chal_p, \Sigma$): Let the challenge be $chal_p = (c, k_1, k_2)$ where $1 \le c \le n$, $k_1 \in \mathcal{Z}_q^*$, $k_2 \in \mathcal{Z}_q^*$. Let F be the set of the blocks. Let Σ be the set of the tags. The hospital queries PCS for medical/health data integrity proof of c file blocks. k_1 is used as the random key of the pseudo-random permutation π. k_2 is used as the random key of the pseudo-random function f. PCS performs the procedures below.

1. For $1 \le i \le c$, PCS computes the indexes and coefficients below: $(I_i, j_i) = \pi_{k_1}(i)$, $a_i = f_{k_2}(i)$
2. The set $\{(I_i, j_i), 1 \le i \le c\}$ is divided into many subsets $\mathcal{S}_{j_i}$ based on the different patients. For the same patient ID_{j_i}, let $\mathcal{S}_{j_i}$ be the subset $\{(I_i, j_i), j_i \textit{ is constant}, 1 \le i \le c\}$. Thus, $\mathcal{S}_{j_i}$ describes the challenged medical/health data blocks of the patient ID_{j_i}. Denote $\mathcal{S} = \{\mathcal{S}_{j_i}, 1 \le i \le c\}$.
   Note: For the different i, maybe, the mapped j_i are the same, *i.e.*, there exist more challenged blocks for the patient ID_{j_i}. Of course, maybe, there doesn't exist challenged blocks for some patients.
3. For $1 \le i \le c$, compute

$$T = \prod_{i=1}^{c} T_{I_i,j_i}^{a_i}, \;\; \hat{F}_{j_i} = \sum_{(I_i,j_i)\in\mathcal{S}_{j_i}} a_i F_{I_i,j_i}$$

4. Denote $\hat{F} = \{\hat{F}_{j_i}, \mathcal{S}_{j_i} \in \mathcal{S}\}$. Output $V = (\hat{F}, T)$ to the hospital.

GenRetr(F, cha_r, Σ): Suppose the patient ID_j wants to retrieve his own medical/health data blocks (I_1, j), $(I_2, j), \cdots, (I_c, j)$. ID_j sends the challenge $chal_r = \{(I_1, j)$, $(I_2, j), \cdots, (I_c, j), k_2\}$ where $k_2 \in \mathcal{Z}_q^*$. Upon receiving the retrieval challenge $chal_r$ from the patient ID_j, PCS performs the procedures below:

1. For $1 \le i \le c$, compute the coefficients: $a_i = f_{k_2}(i)$.
2. For $1 \le i \le c$, compute $T = \prod_{i=1}^{c} T_{I_i,j}^{a_i}$.
3. Output $V_r = (F_{I_1,j}, F_{I_2,j}, \cdots, F_{I_c,j}, T)$ to ID_j.

When the hospital and *AuthSet* cooperate to query PCS to retrieve the patient ID_j's medical/health data, they sends the challenge $chal_r = \{\omega_j, Sign(\omega_j), (I_1, j), (I_2, j), \cdots,$ $(I_c, j), k_2\}$ to PCS. PCS verifies the warrant-signature pair $(\omega_j, Sign(\omega_j))$. If it is valid and the query complies with the warrant ω_j, PCS performs the same procedures as ID_j's retrieval query. Otherwise, rejects it.

CheckProof($y, X_j, Y, Z, chal_p, V_p, ID_j \in \mathcal{P}$): Upon receiving the response V_p from PCS, the hospital performs the procedures below:

1. For $1 \le i \le c$, compute the indexes and coefficients below: $(I_i, j_i) = \pi_{k_1}(i)$, $a_i = f_{k_2}(i)$
2. For $1 \le i \le c$, compute $\hat{t}_{j_i} = H(\hat{e}(X_{j_i}, Z)^y, \omega_{j_i})$;
3. Check whether the following formula holds.

$$\hat{e}(T, g) \overset{?}{=} \prod_{\mathcal{S}_{j_i}\in\mathcal{S}} \hat{e}\left(\prod_{(I_i,j_i)\in\mathcal{S}_{j_i}} h(\Omega_{\hat{t}_{j_i}}(I_i, j_i))^{a_i} u_{j_i}^{\hat{F}_{j_i}}, X_{j_i} \right) \tag{1}$$

If (1) holds, then the hospital outputs "success". Otherwise, the hospital outputs "failure".

Retrieval((x_j, sk_j) *or* $(y, sk_j'$*s valid share set*$), X_j, Y, Z,$ $chal_r, V_r$): The two cases can be considered. (1) The patient ID_j retrieves his own medical/health data. (2) The hospital and $AuthSet_j$ cooperate to retrieve ID_j's medical/health data.

The first case, ID_j retrieves his own medical/health data below:

1. For $1 \le i \le c$, compute the coefficients: $a_i = f_{k_2}(i)$.
2. Compute $\hat{F}_j = \sum_{i=1}^{c} a_i F_{I_i,j}$, $t_j = H(\hat{e}(Y, Z)^{x_j}, \omega_j)$.
3. Check whether the following formula holds.

$$\hat{e}(T, g) \overset{?}{=} \hat{e}\left(\prod_{i=1}^{c} h(\Omega_{t_j}(I_i, j))^{a_i} u_j^{\hat{F}_j}, X_j \right) \tag{2}$$

If the formula (2) holds, the patient ID_j accepts the blocks $F_j' = \{F_{I_1,j}, F_{I_2,j}, \cdots, F_{I_c,j}\}$. Then, the corresponding plaintext $M_j' = D_{sk_j}(F_j')$ can be retrieved by using the symmetric encryption key sk_j. Otherwise, the patient ID_j rejects the response.

The second case, in the authorized set $AuthSet_j$, suppose that th_j entities agree to retrieve the patient ID_j's medical/health data. Let the th_j entities be $D_{j,i_1}, D_{j,i_2}, \cdots, D_{j,i_{th_j}}$. Under the help of the hospital, they cooperate to perform the steps below:

1. For $1 \le i \le c$, compute the coefficients: $a_i = f_{k_2}(i)$. After that, it computes $\hat{F}_j = \sum_{i=1}^{c} a_i F_{I_i,j}$, $t_j = H(\hat{e}(X_j, Z)^y, \omega_j)$.
2. Check whether the following formula holds.

$$\hat{e}(T, g) \overset{?}{=} \hat{e}\left(\prod_{i=1}^{c} h(\Omega_{t_j}(I_i, j))^{a_i} u_j^{\hat{F}_j}, X_j \right) \tag{3}$$

If the formula (3) holds, they accept the blocks $F_j' = \{F_{I_1,j}, F_{I_2,j}, \cdots, F_{I_c,j}\}$. By using their own shares, these th_j entities can compute sk_j:

$$sk_j = \sum_{r=1}^{th_j} \left(\prod_{l=1, l\ne r}^{th_j} \frac{-D_{j,i_l}}{D_{j,i_r} - D_{j,i_l}} \right) ss_{j,i_r}$$

The corresponding plaintext $M_j' = D_{sk_j}(F_j')$ can be retrieved by using the symmetric encryption key sk_j. Otherwise, they reject the response.

3.2 Security Analysis

The correctness analysis and security analysis of our proposed ICS protocol are given by the lemmas and theorems below:

Theorem 1. *If the patient ID_j, hospital and PCS are honest and follow the proposed procedures, then any block-tag pair can pass PCS's tag checking,* i.e.*, CheckTagSign satisfies the correctness.*

Proof. Since (*Sign*, *Verify*) is secure signature-verification algorithm pair, and $(\omega_j, Sign(\omega_j))$ is valid warrant-signature pair, thus, $(\omega_j, Sign(\omega_j))$ can pass the verification. According to the phases of *EncTagGen* and *CheckTagSign*, the following formulas hold:

$$\begin{aligned}
&\hat{t} = H(\hat{e}(X_j, Y)^z, \omega_j) = H(\hat{e}(Y, Z)^{x_j}, \omega_j) = t \\
&\hat{W}_{i,j} = \Omega_{\hat{t}}(i, j) = \Omega_t(i, j) = W_{i,j} \\
&\hat{e}(T_{i,j}, g) = \hat{e}((h(W_{i,j}) u_j^{F_{i,j}})^{x_j}, g) \\
&\qquad = \hat{e}(h(\hat{W}_{i,j}) u_j^{F_{i,j}}, g^{x_j}) = \hat{e}(h(\hat{W}_{i,j}) u_j^{F_{i,j}}, X_j)
\end{aligned}$$

□

Theorem 2. *If hospital and PCS are honest and follow the proposed procedures, the response V_p can pass the hospital's data integrity checking, i.e., CheckProof satisfies the correctness.*

Proof. Let the challenge be $chal_p = (c, k_1, k_2)$. According to the phases of *EncTagGen* and *GenProof*, we know that $\hat{t}_j =$

$H(\hat{e}(X_j, Z)^y, \omega_j) = H(\hat{e}(Y, Z)^{x_j}, \omega_j) = t_j$, $\hat{W}_{i,j} = \Omega_{\hat{t}_j}(i, j) = \Omega_{t_j}(i, j) = W_{i,j}$. Thus,

$$\begin{aligned}
\hat{e}(T, g) &= \hat{e}\left(\prod_{i=1}^{c} T_{I_i, j_i}^{a_i}, g\right) \\
&= \hat{e}\left(\prod_{\mathcal{S}_{j_i} \in \mathcal{S}} \prod_{(I_i, j_i) \in \mathcal{S}_{j_i}} T_{I_i, j_i}^{a_i}, g\right) \\
&= \hat{e}\left(\prod_{\mathcal{S}_{j_i} \in \mathcal{S}} \prod_{(I_i, j_i) \in \mathcal{S}_{j_i}} (h(W_{I_i, j_i}) u_{j_i}^{F_{I_i, j_i}})^{a_i x_{j_i}}, g\right) \\
&= \prod_{\mathcal{S}_{j_i} \in \mathcal{S}} \hat{e}\left(\prod_{(I_i, j_i) \in \mathcal{S}_{j_i}} (h(\hat{W}_{I_i, j_i}) u_{j_i}^{F_{I_i, j_i}})^{a_i}, g^{x_{j_i}}\right) \\
&= \prod_{\mathcal{S}_{j_i} \in \mathcal{S}} \hat{e}\left(\prod_{(I_i, j_i) \in \mathcal{S}_{j_i}} h(\Omega_{\hat{t}_{j_i}}(I_i, j_i))^{a_i} u_{j_i}^{\hat{F}_{j_i}}, X_{j_i}\right)
\end{aligned}$$

□

Theorem 3. *If the patient ID_j (or hospital and $AuthSet_j$) and PCS are honest and follow the proposed procedures, ID_j (or hospital and $AuthSet_j$) can retrieve the queried medical/health data,* i.e., Retrieval *satisfies the correctness.*

Proof. When the patient ID_j queries to retrieve his own medical/health data, the verification formula (2) holds based on the theorem 2. Then, it is straightforward to get the medical/health data by decrypting the ciphertext using his own symmetric encryption key sk_j.

When the hospital and $AuthSet_j$ cooperate to retrieve ID_j's medical/health data, by Lagrange interpolation formula, the sk_j can be obtained below:

$$sk_j = \sum_{r=1}^{th_j} \left(\prod_{l=1, l \neq r}^{th_j} \frac{-D_{j,i_l}}{D_{j,i_r} - D_{j,i_l}} \right) ss_{j,i_r}$$

After that, since $t_j = H(\hat{e}(X_j, Z)^y, \omega_j) = H(\hat{e}(Y, Z)^{x_j}, \omega_j)$, they can also get ID_j's medical/health data by performs the similar procedures as the patient ID_j. □

Theorem 4 (Possession Against Malicious PCS)**.** *On the GDH group $\mathcal{G}_1$, based on the difficulty of CDH problem, the proposed ICS protocol is existentially unforgeable in the random oracle model. That is, the proposed ICS protocol satisfies the security property of provable data integrity against malicious PCS.*

Proof. It is similar with the Ref. [10]. We omit it due to the page limits. □

Theorem 5 (Restrictive Proof of Possession)**.** *For the remote medical/health data, the proposed ICS protocol satisfies restrictive proof of integrity.*

Proof. From the theorem 2, the hospital can perform all the patients' medical/health data integrity checking. For the patient ID_j, he can compute the parameter t_j and $W_{i,j}$. Based on the two parameters, ID_j can perform the proof of his own medical/health data integrity below: $\hat{e}(T, g) \overset{?}{=} \hat{e}(\prod_{(I_i, j) \in \mathcal{S}_j} h(\Omega_{\hat{t}_j}(I_i, j))^{a_i} u_j^{\hat{F}_j}, X_j)$.

Except for the hospital, the patients and PCS, the third party has no ability to get $\hat{t}_j$ based on the difficulty of BDHP. Thus, the third party can not also compute $\Omega_{\hat{t}_{j_i}}(I_i, j_i)$. Finally, the third party can not perform the verification equation:

$$\hat{e}(T, g) \overset{?}{=} \prod_{\mathcal{S}_{j_i} \in \mathcal{S}} \hat{e}\left(\prod_{(I_i, j_i) \in \mathcal{S}_{j_i}} h(\Omega_{\hat{t}_{j_i}}(I_i, j_i))^{a_i} u_{j_i}^{\hat{F}_{j_i}}, X_{j_i}\right)$$

Thus, the proposed ICS protocol satisfies restrictive proof of integrity. □

Theorem 6 (Restrictive Retrievability)**.** *For the remote medical/health data, the proposed ICS protocol satisfies the property of restrictive retrievability.*

Proof. From the theorem 3, the patients have the ability to retrieve their own medical/health data. In the patient ID_j's authorized entities, if at least th_j entities agree to retrieve ID_j's remote medical/health data, these authorized entities and the hospital have the ability to cooperate to retrieve ID_j's remote medical/health data.

On the contrary, if less than $th_j - 1$ authorized entities agree to retrieve ID_j's data, they only succeed with negligible probability. According to the process of symmetric encryption key distribution, the function $\mathcal{F}_j$ has the order $th_j - 1$. It can be determined by at least th_j points. Less than $th_j - 1$ authorized entities can provide less than $th_j - 1$ points. $\mathcal{F}_j$ can not be determined and the symmetric encryption key sk_j can not also be determined. Thus, they have no ability to retrieve ID_j's remote medical/health data. □

Theorem 7. *The proposed ICS protocol is $\left(\frac{d}{n}, 1 - \left(\frac{n-d}{n}\right)^c\right)$-secure since the probability P_R of detecting the corruption satisfies:*

$$1 - \left(\frac{n-d}{n}\right)^c \leq P_R \leq 1 - \left(\frac{n-c+1-d}{n-c+1}\right)^c$$

where PCS has stored $n = n_1 + n_2 + \cdots + n_{n_p}$ block-tag pairs for n_p patients, PCS has corrupted d block-tag pairs, and the challenge is $chal_p = (c, k_1, k_2)$.

Proof. It is similar with Ref. [9]. We omit it due to the page limits. □

4. Performance

We implemented our ICS scheme in order to demonstrate the effectiveness of our scheme. We used the C programming language with the GMP (GMP-5.0.5), Miracl and PBC (pbc-0.5.13) libraries. In the implementation, PCS ran on the laptop with the following features:

- CPU: Intel Core i7-3517U @ 1.90GHz
- Physical Memory: 4GB DDR3 1600MHz

- OS: Ubuntu 13.04 Linux 3.8.0-19-generic SMP i686

Hospital, Patient and *AuthSet* ran on a laptop with the following features:

- CPU: CPU I PDC E6700 3.2GHz
- Physical Memory: DDR3 2G
- OS: Ubuntu 11.10 over VMware-workstation-full-8.0.0

Our cryptographic choices were: i) an elliptic curve with 160-bit group order; and iii) AES (Advanced Encryption Standard) as the symmetric encryption algorithm. Figure 1 depicts PCS's computation time cost in the phase of *GenProof* and *GenRetr*. In the X-axis we represent the number c of challenged blocks. The Y-axis represents PCS's computation time in ms (*i.e.*, milliseconds) in order to generate the proof or retrieve the block. Figure 2 depicts hospital computation time cost in *CheckProof*. The X-axis represents the number c of challenged blocks. The Y-axis represents hospital's computation time (s) to check the proof, where $p1$, $p2$, $p3$ denotes the challenged patients' number. Thus, the challenged block number must be bigger than the challenged patients, *i.e.*, $p1 \geq c$, $p2 \geq c$, $p3 \geq c$. Figure 3 depicts the time cost of hospital and AuthSet in the phase *Retrieval*. The X-axis represents the number c of challenged blocks. The Y-axis represents hospital and AuthSet's computation time (ms) in order to retrieve c blocks in the phase *Retrieval*.

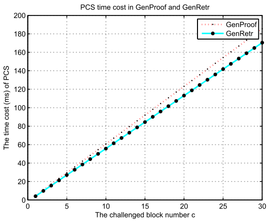


Fig. 1 PCS time cost in GenProof and GenRetr.

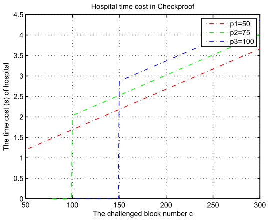


Fig. 2 Hospital time cost in CheckProof.

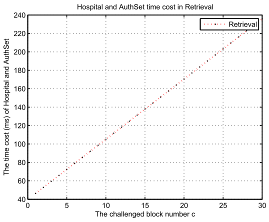


Fig. 3 Hospital and AuthSet time cost in Retrieval.

5. Conclusions

In this paper, we propose the concept of ICS protocol for critical patients in cloud-based health internet of things. This paper formalizes the system model and security model of ICS protocol. Based on the pairing, a concrete ICS protocol is designed. The proposed ICS protocol is provably secure and efficient by security analysis and performance analysis.

Acknowledgments

The work of H. Wang was supported in part by the National Natural Science Foundation of China under Grant (61272522), in part by the Natural Science Foundation of Liaoning Province under Grant (2014020147), and in part by the Program for Liaoning Excellent Talents in University under Grant (LR2014021), and in part by the CICAEET fund and the PAPD fund. This work of K. Li was partly supported by the National Science Foundation for Distinguished Young Scholars of China (61225010), and the State Key Program of National Natural Science of China (61432002).

References

[1] F. Hu, D. Xie, and S. Shen, "On the application of the internet of things in the field of medical and health care," IEEE International Conference on Cyber, Physical and Social Computing, pp.2053–2058, 2013.

[2] W. Zao, C. Wang, and Y. Nakahira, "Medical application on internet of things," ICCTA 2011, pp.660–665, 2011.

[3] D. He and S. Zeadally, "An analysis of RFID authentication schemes for internet of things in healthcare environment using elliptic curve cryptography," IEEE Internet Things J., vol.2, no.1, pp.72–83, 2014.

[4] D. He, N. Kumar, J. Chen, C.-C. Lee, N. Chilamkurti, and S.S. Yeo, "Robust anonymous authentication protocol for health-care applications using wireless medical sensor networks," Multimedia Systems, vol.21, no.1, pp.49–60, 2015.

[5] J. Wu, M. Dong, K. Ota, L. Liang, and Z. Zhou, "Securing distributed storage for social internet of things using regenerating code and Blom key agreement," Peer-to-Peer Networking and Applications, vol.8, no.6, pp.1133–1142, 2015.

[6] H. Wang, Q. Wu, B. Qin, and J. Domingo-Ferrer, "FRR: Fair remote retrieval of outsourced private medical records in electronic health networks," Journal of Biomedical Informatics, vol.50, pp.226–233, 2014.

[7] R. Lu, X.D. Lin, and X.M. Shen, "SPOC: A secure and privacy-preserving opportunistic computing framework for mobile-healthcare emergency," IEEE Trans. Parallel Distrib. Syst., vol.24, no.3, pp.614–624, 2013.

[8] J. He, M. Dong, K. Ota, M. Fan, and G. Wang, "NetSecCC: A scalable and fault-tolerant architecture for cloud computing security," Peer-to-Peer Networking and Applications, vol.9, no.1, pp.67–81, 2016.

[9] G. Ateniese, R. Burns, R. Curtmola, J. Herring, L. Kissner, Z. Peterson, and D. Song, "Provable data integrity at untrusted stores," CCS '07, pp.598–609, 2007.

[10] H. Wang, "Proxy provable data possession in public clouds," IEEE Trans. Services Computing, vol.6, no.4, pp.551–559, 2013.

[11] K.D. Bowers, A. Juels, and A. Oprea, "Proofs of retrievability: Theory and implementation," CCSW '09, pp.43–54, 2009.

[12] Y. Ren, J. Shen, J. Wang, J. Han, and S. Lee, "Mutual verifiable provable data auditing in public cloud storage," Journal of Internet Technology, vol.16, no.2, pp.317–323, 2015.

[13] Z. Xia, X. Wang, X. Sun, and Q. Wang, "A secure and dynamic multi-keyword ranked search scheme over encrypted cloud data," IEEE Trans. Parallel Distrib. Syst., vol.27, no.2, pp.340–352, 2016.

[14] Z. Fu, X. Sun, Q. Liu, L. Zhou, and J. Shu, "Achieving efficient cloud search services: Multi-keyword ranked search over encrypted cloud data supporting parallel computing," IEICE Trans. Commun., vol.E98-B, no.1, pp.190–200, Jan. 2015.

[15] C. Wang, S.S.M. Chow, Q. Wang, K. Ren, and W. Lou, "Privacy-preserving public auditing for secure cloud storage," IEEE Trans. Comput., vol.62, no.2, pp.362–375, 2013.

[16] M. Nabeel, N. Shang, and E. Bertino, "Privacy preserving policy based content sharing in public clouds," IEEE Trans. Knowl. Data Eng., vol.25, no.11, pp.2602–2614, 2013.

[17] P. Guo, J. Wang, X.H. Geng, and J.U. Kim, "A variable threshold-value authentication architecture for wireless mesh networks," Journal of Internet Technology, vol.15, no.6, pp.929–935, 2014.

[18] D. Boneh and M. Franklin, "Identity-based encryption from the Weil pairing," Advances in Cryptology, CRYPTO '01, Lecture Notes in Computer Science, vol.2139, pp.213–229, Springer, Berlin, Heidelberg, 2001.

[19] A. Miyaji, M. Nakabayashi, and S. Takano, "New explicit conditions of elliptic curve traces for FR-reduction," IEICE Trans. Fundamentals, vol.E84-A, no.5, pp.1234–1243, May 2001.

[20] D. Boneh, B. Lynn, and H. Shacham, "Short signatures from the Weil pairing," Advances in Cryptology, ASIACRYPT '01, Lecture Notes in Computer Science, vol.2248, pp.514–532, Springer, Berlin, Heidelberg, 2001.

[21] B. Lynn, "The pairing-based cryptography library," http://crypto.stanford.edu/pbc/times.html

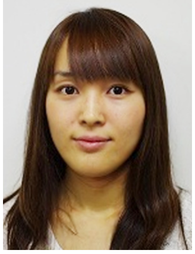
[22] R. Kumanduri, Number Theory with Computer Applications, Prentice Hall, Upper Saddle River, NJ, USA, 1998.

[23] V. Miller, "Uses of elliptic curves in cryptography," Advances in Cryptology, CRYPTO '85, Lecture Notes in Computer Science, vol.218, pp.417–426, Springer, Berlin, Heidelberg, 1985.

[24] S. Vanstone, "Responses to NISTs proposal," Commun. ACM, vol.35, pp.50–52, 1992.

Huaqun Wang received the BS degree in mathematics education from the Shandong Normal University, Jinan, China, in 1997, the MS degree in applied mathematics from the East China Normal University, Shanghai, China, in 2000, and the PhD degree in cryptography from Nanjing University of Posts and Telecommunications, Nanjing, China, in 2006. Since then, he has been with Dalian Ocean University, Dalian, China, as a Full Professor. His research interests include applied cryptography, network security, and cloud computing security. Dr. Wang has published more than 50 papers. He has served in the program committee of several international conferences and the editor board of international journals.

Keqiu Li received the bachelor's and master's degrees from the Department of Applied Mathematics, Dalian University of Technology in 1994 and 1997, respectively. He received the PhD degree from the Graduate School of Information Science, Japan Advanced Institute of Science and Technology in 2005. He also has a two-year postdoctoral experience in the University of Tokyo, Japan. He is currently a professor in the School of Computer Science and Technology, Dalian University of Technology, China. He has published more than 100 technical papers, such as IEEE TPDS, ACM TOIT, and ACM TOMCCAP. He is an associate editor of IEEE TPDS and IEEE TC. His research interests include internet technology, data center networks, cloud computing and wireless networks. He is a senior member of the IEEE.



Kaoru Ota received M.S. degree in Computer Science from Oklahoma State University, USA in 2008 and Ph.D. degree in Computer Science and Engineering from The University of Aizu, Japan in 2012. She is currently an Assistant Professor with Department of Information and Electronic Engineering, Muroran Institute of Technology, Japan. From March 2010 to March 2011, she was a visiting scholar with BBCR group at University of Waterloo, Canada. Also she was a Japan Society of the Promotion of Science (JSPS) research fellow with Kato-Nishiyama Lab at Graduate School of Information Sciences at Tohoku University, Japan from April 2012 to April 2013. She has joined JSPS A3 foresight program as one of primary researchers since 2011 which is supported by Japanese, Chinese and Korean government. Dr. Ota's research results have been published in 90 research papers in international journals, conferences and books. She is the Best Paper Award Winner of ICA3PP 2014, GPC 2015 and IEEE DASC 2015. She serves a Guest Editor of IEEE Wireless Communications, IEICE Transactions on Information and Systems and serves Editor of Peer-to-Peer Networking and Applications (Springer), Ad Hoc & Sensor Wireless Networks, International Journal of Embedded Systems (Inderscience). Her research interests include wireless sensor networks, vehicular ad hoc networks, and ubiquitous computing.

Jian Shen received the B.E. degree from Nanjing University of Information Science and Technology, Nanjing, China, in 2007 and the M.E. and Ph.D. degrees in Computer Science from Chosun University, Gwangju, Korea, in 2009 and 2012, respectively. Since late 2012, he has been a faculty member in the School of Computer and Software at Nanjing University of Information Science and Technology, Nanjing, China. His research interests include computer networking, security systems, mobile computing and networking, ad-hoc networks and systems, and ubiquitous sensor networks.