IJSEAT International Journal of Science Engineering and Advance Technology

Discretion Preservative Itinerant Health Scrutinising By Haze Succour

T Jeevana Jyothi#1,**Sk Mubeena Sultana**#2
#1student Of M.Tech (C.S) And Department of Cse,G.V.R&Scollege Of Engineering &Technology.
#2 Department of Cse, G.V.R&Scollege Of Engineering & Technology

Abstract—Haze-assisted mobile health (mHealth) monitoring,which applies the prevailing mobile communications and Haze computing technologies to provide feedback decision support, has been considered as a revolutionary approach to improving the quality of healthcare service while lowering the healthcare cost. Unfortunately, it also poses a serious risk on both clients Confidentiality and intellectual property of monitoring service providers, which could deter the wide adoption of mHealth technology. This paper is to address this important problem and design a Hazeassisted Confidentiality Conserving mobile health monitoring system to protect the Confidentiality of the involved parties and their data. Moreover, the outsourcing decryption technique and a newlyproposed key private proxy re-encryption are adapted to shift the computational complexity of the involved parties to the Haze without compromising clients' Confidentiality and service providers' intellectual property. Finally, our security and performance analysis demonstrates the effectiveness of our proposed design.

Index Terms—Mobile health (mHealth), Healthcare,Confidentiality, Outsourcing decryption, Key private proxy reencryption.

I. INTRODUCTION

Wide deployment of mobile devices, such as smart phones equipped with low cost sensors, has already shown great potential in improving the quality of healthcare services. Remote mobile health monitoring has already been recognized as not only a potential, but also a successful example of mobile health (mHealth) applications especially for developing countries. The Microsoft launched project "MediNet" is designed to realize remote monitoring on the health status of diabetes and cardiovascular diseases in remote areas in Caribbean countries [1]. In such a remote mHealth monitoring system, a client could deploy portable sensors in wireless body sensor networks to collect various physiological data, such as blood pressure (BP), breathing rate (BR), Electrocardiogram (ECG/EKG), peripheral oxygen saturation (SpO2) and blood glucose. Such physiological data could then be sent to a central server, which could then run various web medical applications on these data to return timely advice to the client. These applications may have various functionalities ranging from sleep pattern analyzers, exercises, physical activity assistants, to cardiac analysis systems, providing various medical consultation [2]. Moreover, as the emerging Haze computing technologies evolve, a viable solution can be sought by incorporating the software as a service (SaaS) model and pay-as-you-go business model in Haze computing, which would allow small companies (healthcare service providers) to excel in this healthcare market. It has been observed that the adoption of automated decision support algorithms in the Haze-assisted mHealth monitoring has been considered as a future trend [3].

Unfortunately, although Haze-assisted mHealth monitoring could offer a great opportunity to improve the quality of healthcare services and potentially reduce healthcare costs, there is a stumbling block in making this technology a reality. Without properly addressing the data management in an mHealth system, clients' Confidentiality may be severely breached during the collection, storage, diagnosis, communications and computing.

A recent study shows that 75% Americans consider the Confidentiality of their health information important or very important [4]. It has also been reported [5] that patients' willingness to get involved in health monitoring program could be severely lowered when people are concerned with the Confidentiality breach in their voluntarily submitted health data. This Confidentiality concern will be exacerbated due to the growing trend in Confidentiality breaches on electronic health data.

Although the existing Confidentiality laws such as HIPAA (Health Insurance Portability and Accountability Act) provide baseline protection for personal health record, they are generally considered not applicable or transferable to Haze computing environments [6]. Besides, the current law is more focused on protection against adversarial intrusions while there is little effort on protecting clients from business collecting private information. Meanwhile, many companies have significant commercial interests in collecting clients' private health data [7] and sharing them with either insurance companies, research institutions or even the government agencies. It has also been indicated [8] that Confidentiality law could not really exert any real protection on clients' data Confidentiality unless there is an effective mechanism to enforce restrictions on the activities of healthcare service providers.

In this paper, we design a Haze-assisted mHealth monitoring system (CAM). We first identify the design problems on Confidentiality preservation and then provide our solutions. To ease the understanding, we start with the basic scheme so that we can identify the possible Confidentiality breaches. We then provide an improved scheme by addressing the identified Confidentiality problems. The resulting improved scheme allows the mHealth service provider (the company) to be offline after the setup stage and enables it to deliver its data or programs to the Haze securely. To reduce clients' decryption complexity, we incorporate the recently proposed outsourcing decryption technique [25] into the underlying multidimensional range queries system to shift clients' computational complexity to the Haze without revealing any information on either clients' query input or the decrypted decision to the Haze. To relieve the computational complexity on the company's side, which is proportional to the number of clients, we propose a further improvement, leading to our final scheme. It is based on a new variant of key private proxy re-encryption scheme, in which the company only needs to accomplish encryption once at the setup phase while shifting the rest computational tasks to the Haze without compromising Confidentiality, further reducing the computational and communication burden on clients and the Haze.

II. SYSTEM MODEL AND ADVERSARIAL MODEL

To facilitate our discussion, we first elaborate our Hazeassisted mHealth monitoring system (CAM). CAM consists of four parties: the Haze server (simply the Haze), the company who provides the mHealth monitoring service (i.e., the healthcare service provider), the individual clients (simply clients), and a semi-trusted authority (TA). The company stores its encrypted monitoring data or program in the Haze server. Individual clients collect their medical data and store them in their mobile devices, which then transform the data into attribute vectors. The attribute vectors are delivered as inputs to the monitoring program in the Haze server through a mobile (or smart) device. A semitrusted authority is responsible for distributing private keys to the individual clients and collecting the service fee from the clients according to a certain business model such as pay-as-you-go business model. The TA can be considered as a collaborator or a management agent for a company (or several companies) and thus shares certain level of mutual interest with the company. However, the company and TA could collude to obtain private health data from client input vectors. We assume a neutral Haze server, which means it neither colludes with the company nor a client to attack the other side. This is a reasonable model since it would be in the best business interest of the Haze not to be biased. We admit that it remains possible for the Haze to collude with other malicious entities in our CAM, and we leave the CAM design under these stronger models as future work. We also do not assume that an individual client colludes with other clients. Our security model does not consider the possible sidechannel attack [26], [27] due to the coresidency on shared resources either because it could be mitigated with either system level protection [27] or leakage resilient cryptography [28]. CAM assumes an honest but curious model, which implies all parties should follow the prescribed actions and cannot be arbitrarily malicious.

In the following, we briefly introduce the four major steps of CAM: Setup, Store, TokenGen and Query. We only illustrate the functionality of these components in this section while leaving the details in later sections.

At the system initialization, TA runs the Setup phase and publishes the system parameters. Then the company first expresses the flow chart of the mHealth monitoring program as a branching program (see Sec. III-B for detail), which is encrypted under the respective directed branching tree. Then the company delivers the resulting ciphertext and its company index to the Haze, which corresponds to the Store algorithm in the context.

When a client wishes to query the Haze for a certain mHealth monitoring program, the *i*-th client and TA run the TokenGen algorithm. The client sends the company index to TA, and then inputs its private query (which is the attribute vector representing the collected health data) and TA inputs the master secret to the algorithm. The client obtains the token corresponding to its query input while TA gets no useful information on the individual query.

During the last phase, the client delivers the token for its query to the Haze, which runs the Query phase. The Haze completes the major computationally intensive task for the client's decryption and returns the partially decrypted ciphertext to the client. The client then completes the remaining decryption task after receiving the partially decrypted ciphertext and obtains its decryption result, which corresponds to the decision from the monitoring program on the clients' input. The Haze obtains no useful information on either the client's private query input or decryption result after running the Query phase. Here, we distinguish the query input Confidentiality breach in terms of what can be inferred from the computational or communication information. CAM can prevent the Haze from deducing useful information from the client's query input or output corresponding to the received information from the client. However, the Haze might still be able to deduce side information on the client's private query input by observing the client's access pattern. This issue could be resolved by oblivious RAM technique [29], but this is out of the scope of this paper.

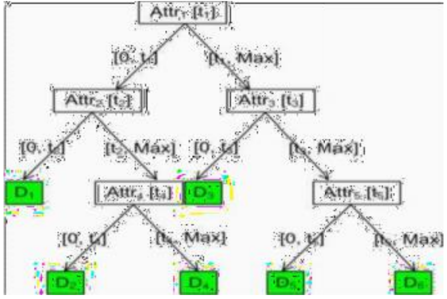

Fig. 1. Branching program

IV. CAM DESIGN

We are ready to present our design *CAM: HazeassistedConfidentiality Conserving mHealth monitoring system*. Toillustrate the fundamental idea behind this design, we start with the basic scheme, and then demonstrate how improvements can be made step-by-step to meet our design goal. Some of the variables in the following illustration may have already been defined in the previous sections. The system time is divided into multiple time periods, called *slots*, each of which can last a week or a month depending on specific application scenarios. There is an estimated maximum number of users N requesting access to the monitoring program in any given slot. When a client attempts to access the program, it is assigned an index $i \in [1,N]$ by TA.

A. Basic CAM

The following basic scheme runs the BF-IBE system as a subroutine and is the fundamental building block in our overall design.

Setup: This algorithm is performed by TA, which publishes the system parameters for the BF-IBE scheme.

Store: This algorithm is performed by the company. For each node pj whose child nodes are not leaf nodes, the company runs $CL(j)$=AnonEnc(id,PP,$L(j)$)and $CR(j)$= AnonEnc(id,PP,$R(j)$)toencrypt the child node indices under id with either id $\in S[0,tj]$ or id $\in S[tj+1,Max]$, respectively. When the child nodes of pj are leaf nodes, the company generates the ciphertext as $CL(j)$ = AnonEnc(id,PP,$mL(j)$) and $CR(j)$ = AnonEnc(id,PP,$mR(j)$), where $mL(j)$and $mR(j)$denote the attached information at the two leafnodes, respectively. All the generated ciphertexts are delivered and stored in the Haze.

TokenGen: To generate the private key for the attribute vector v=($v1,\cdots,vn$), a client first computes the identity representation set of each element in v and delivers all the n identity representation sets to TA. Then TA runs the AnonExtract(id,msk) on each identity id $\in Svi$ in the identity set and delivers all the respective private keys skvi to the client.

Query: A client delivers the private key sets obtained from the TokenGen algorithm to the Haze, which runs the AnonDecryption algorithm on the ciphertext generated in the Store algorithm. Starting from p1, the decryption result determines which ciphertext should be decrypted next. For instance, if $v1 \in [0,t1]$, then the decryption result indicates the next node index $L(i)$. The Haze will then use sk$v(L(i))$ to decrypt the subsequent ciphertext $CL(i)$. Continue this process iteratively until it reaches a leaf node and decrypt the respective attached information.

B. CAM with Full Confidentiality Preservation

The basic scheme has the following security weakness: first, the identity representation set for a client's attribute vector v is known to TA, and hence TA can easily infer all the client's private attribute vector. Second, the client cannot protect his Confidentiality from the Haze either because the Haze can easily find out the identity representation for the private key sk$vi,i \in [1,n]$ by running identity test in MDRQs. The Haze can simply encrypt a random message under any attribute value v' until when it can use skvi to successfully decrypt the ciphertext, which means there is a match between $v^{'} = vi$ and hence it successfully finds out *vi*. Third, neither can the data Confidentiality of the company be guaranteed since the identity representation of the respective range is revealed to the Haze whenever the decryption is successful due to the match revealing property (see Sec. III-D) of MDRQs. The Haze can finally figure out most of the company's branching program since it has the private keys of all the system users.

To rectify the weakness of the basic scheme, we provide the following improvement. The high level idea (as shown in Fig. 4) is as follows: in order to avoid leaking the attribute vector to TA, the client obliviously submits his attribute vectors to TA so that he can obtain the respective private keys without letting TA get any useful information on his private vector. The client runs the outsourcing decryption of MDRQs to ensure the Haze completes the major workload while obtaining no useful information on his private keys. On the other hand, the company will permute and randomize its data using homomorphic encryption and MDRQs so that neither the Haze nor a client can get any useful information on its private information on branching program after a single query.

Meanwhile, the company is also required to include the randomness in the randomization step in the encryption sent to TA to guarantee that TA can successfully generate tokens for clients.

C. Final CAM with Full Confidentiality and High Efficiency

Although the above improved scheme does meet the desired security requirements, the company may need to compute all the ciphertexts for each of N clients, which implies huge $^{\text{algorith}}$ m and computational overheads and may not be economically feasible for small mHealth companies. In this section, we provide a further improvement to reduce both the computational burden on the company and the communication overhead for the Haze. The high level idea (as shown in Fig. 5) is as follows. We employ a newly developed key private re-encryption scheme (introduced in Sec. IV-C1) as an underlying tool. Instead of computing a ciphertext for each client, the company generates one single ciphertext, which will then be delivered to the Haze. The company will then obliviously deliver the identity threshold representation sets for the thresholds of the decisional branching nodes and the indexes of the concerned attributes to TA so that TA can generate the ReKeys corresponding to the rest clients in the system using the key private reencryption scheme. The generated rekeys are then delivered to the Haze, which can then run the reencryption scheme using the rekeys and the single ciphertext delivered by the company to generate the ciphertexts for the rest clients. The proposed reencryption scheme incorporates the outsourcing decryption so that the other security and efficiency characteristics in the final CAM are inherited here. Besides, the decryption algorithm of the proxy reencryption scheme induces much less interactions between clients and the Haze comparing with that in our improved scheme.

VI. CONCLUSION

In this paper, we design a Haze-assisted Confidentiality Conserving mobile health monitoring system, called CAM, which can effectively protect the Confidentiality of clients and the intellectual proerty of mHealth service providers. To protect the clients' Confidentiality, we apply the anonymous BonehFranklin identitybased encryption (IBE) in medical diagnostic branching programs. To reduce the decryption complexity due to the use of IBE, we apply recently proposed decryption outsourcing with Confidentiality protection to shift clients' pairing computation to the Haze server. To protect mHeath service providers' programs, we expand the branching program tree by using the random permutation and randomize the decision thresholds used at the decision branching nodes. Finally, to enable resourceconstrained small companies to participate in mHealth business, our CAM design helps them to shift the computational burden to the Haze by applying newly developed key private proxy re-encryption technique. Our CAM has been shown to achieve the design objective.

REFERENCES

[1] P. Mohan, D. Marin, S. Sultan, and A. Deen, "Medinet: personalizing the self-care process for patients with diabetes and cardiovascular disease using mobile telephony." *Conference Proceedings of the InternationalConference of IEEE Engineering in Medicine and BiologySociety*, vol.2008, no. 3, pp. 755–758. [Online].Available:http://www.ncbi.nlm.nih.gov/pubmed/1916 2765

[2] A. Tsanas, M. Little, P. McSharry, and L. Ramig,"Accurate telemonitoring of parkinson's disease progression by noninvasive speech tests,"*Biomedical Engineering, IEEE Transactions on*, vol. 57, no. 4, pp. 884 – 893, 2010.

[3] G. Clifford and D. Clifton, “Wireless technology in disease management and medicine,” *Annual Review of Medicine*, vol. 63, pp. 479 – 492, 2012.

[4] L. Ponemon Institute,“Americans’ opinions on healthcare Confidentiality, available: http://tinyurl.com/4atsdlj,” 2010.

[5] A. V. Dhukaram, C. Baber, L. Elloumi, B.J. van Beijnum, and P. D. Stefanis, “End-user perception towards pervasive cardiac healthcare services: Benefits, acceptance, adoption, risks, security, Confidentiality and trust,” in*PervasiveHealth*, 2011, pp. 478–484.

[6] M. Delgado, “The evolution of health care it: Are current u.sConfidentiality policies ready for the Hazes?” in *SERVICES*, 2011, pp. 371 –378.

[7] N. Singer, “When 2+ 2 equals a Confidentiality question,” *New YorkTimes*, 2009.

[8] P. Baldi, R. Baronio, E. D. Cristofaro, P. Gasti, and G. Tsudik,“Countering gattaca: efficient and secure testing of fully-sequenced human genomes,” in *ACM Conference on Computer and Communications Security*, 2011, pp. 691–702.

[9] A. Cavoukian, A. Fisher, S. Killen, and D. Hoffman, “Remote home health care technologies: how to ensure Confidentiality? build it in: Confidentiality by design,” *Identity in the Information Society*, vol. 3, no. 2, pp. 363–378, 2010.

[10] A. Narayanan and V. Shmatikov, “Robust de-anonymization of large sparse datasets,” in *Security and Confidentiality, 2008. SP 2008. IEEESymposium on*. IEEE, 2008, pp. 111–125.

[11] ——, “De-anonymizing social networks,” in *IEEE Symposium on Security and Confidentiality*. IEEE Computer Society, 2009, pp. 173–187.

[12] I. Neamatullah, M. Douglass, L. Lehman, A. Reisner, M. Villarroel, W. Long, P. Szolovits, G. Moody, R. Mark, and G. Clifford, “Automated deidentification of freetext medical records,” *BMC medical informatics and decision making*, vol. 8, no. 1, p. 32, 2008.

[13] S. Al-Fedaghi and A. Al-Azmi, “Experimentation with personal identifiable information,” *Intelligent Information Management*, vol. 4, no. 4 , pp. 123–133, 2012.

[14] J. Domingo-Ferrer, “A three-dimensional conceptual framework for database Confidentiality,” *Secure Data Management*, pp. 193–202, 2007.

[15] T. Lim, *Nanosensors: Theory and Applications in Industry, Healthcare,and Defense*. CRC Press, 2011.

[16] X. Zhou, B. Peng, Y. Li, Y. Chen, H. Tang, and X. Wang, “To release or not to release: evaluating information leaks in aggregate human-genome data,” *Computer Security–ESORICS 2011*, pp. 607–627, 2011.

AUTHORS:

T JEEVANA JYOTHI is a student of computer science engineering from G.V.R&Scollege of engineering &technology, presently pursuing m.tech (cse) from this college. She received B.Tech from JNTUK in the year of 2012.

SK MUBEENA SULTANA is a ASSISTANT PROFESSOR from G.V.R&S College of Engineering &Technology. She received M.Tech from JNTU Kakinada University. She gained a teaching experience of 3 Years.she is good researcher in Networking.