Secure Mobile Cloud Architecture for Healthcare Application

Nacha Chondamrongkul and Pattra Chondamrongkul

***Abstract*—Demand of healthcare service has been increasing worldwide. With advance development of pervasive technology, wearable device with multiple sensors can be effectively used as instruments to remotely supervise patient condition at any place and any time. However, this requires dynamically scale storage resources to handle tremendous amount of data that generated from various sensors. So the public cloud suits this requirement very well, as it provides high scalable data storage service with effective cost. But data confidentiality and privacy are still challenging issues, because cloud infrastructure is maintained by the third party, so data is security compromised and generally prone to potential treats. We therefore propose architecture with security schemes that aim to overcome this challenge. The architecture supports mobile application that continuously monitors patient health condition in the secure way. The security scheme is based on public key infrastructure to protect confidentiality, privacy and authenticity of personal data. Our proposed architecture has been implemented and initially tested by group of sampling patients at a local hospital in Thailand.**

***Index Terms*—Healthcare, cloud computing, mobile computing, public key infrastructure, security.**

I. Introduction

The rapid growth of using wearable device and smart phone play a significant role in accelerating the development of healthcare application. As smart phone and wearable device are consisted of various sensors that can detect heart pulse, respiratory and blood pressure, so continuous remote monitoring can be achieve with minimum cost, and enable healthcare service provider to give effective treatment to the patient. Number of sensors periodically collect data which are later remotely transmitted to store on server hosted by healthcare provider for further analysis. As sensors generate data with exponential growth, how to handle tremendous amount of data is a challenge as it require high scalability infrastructure to support. Cloud computing can overcome this challenge, however there is security vulnerabilities especially on public cloud that the infrastructure is maintained by service provider's staff. Even though service provider cannot legally access these data owned by their customer, but these data is vulnerable to malicious program that allow data to be accessed without authorization. Because health and medical data is highly sensitive, so security and privacy becomes problems. This paper propose an architecture with security scheme that help eliminating potential security treats of health and medical data situated on the cloud. The architecture serves mobile application that support authorized doctors and medical staff to remotely monitor patient in secure manner.

The rest of this paper is organized as follows. Section II addresses related works, highlight existing approach and methods in securing application in healthcare domain. We explain proposed architecture and security scheme in section III and IV respectively. Section V presents the illustration of proposed architecture with state of the art technologies. The paper concludes in Section VI.

Manuscript received December 12, 2016; revised April 7, 2017.

The authors are with Chiang Rai Rajabhat University, Thailand (e-mail: nacha.cho@mfu.ac.th; p.pattra1985@gmail.com).

II. Related Work

As healthcare system is required to be secure, there are many proposals on how to protect confidential and privacy of information on the cloud. Cloud computing provides scalability and availability to the software system, however it pose a high risk that information stored on infrastructure of cloud service provider can be access by other unauthorized entity or malicious program. Cryptographic technique is widely proposed for securing healthcare system on the cloud with both symmetric and asymmetric encryption algorithm. The main challenge is how keys and encryption are smoothly managed without affecting usability of the system, and allow only authorized entity to access only information they need. Therefore, fine grain access control becomes a focus. Goyal *et al*. [1] propose Attribute-Based Encryption (ABE) for fine-grained access control and Kumar *et al*. [2] enhance ABE to be effective for cloud critical application. Therefore, ABE can also be applied to secure healthcare system on the cloud, however there are challenges such as how access control can be managed, and integration of key and access structure. Therefore, Lounis *et al*. [3] propose fine grain access control that combines ABE and symmetric cryptography to tackle these problems for cloud-based healthcare system. Barua *et al*. [4] proposes new security scheme called ESPAC that can handle access control problem on personal health information based on ABE. There are also other works [5]-[7] that apply ABE with healthcare system. Beside, identity-based encryption [8] is also proposed to integrate into healthcare system, as it uses the user's identity information as public key and can be cost-effective on the cloud such as a work presented by Wanga *et al*. [9].

III. Proposed Architecture

The proposed architecture supports healthcare system that enables patient to be monitored by mobile applications. Personal record application helps gathering health data from

doi: 10.18178/ijfcc.2017.6.3.493

connected wearable devices and smart phone, before storing them on the cloud. After that, monitoring application retrieve these data to enables doctor and involved medical staffs to supervise patient's condition. We design security scheme based on Public Key Infrastructure (PKI) [10] and RSA algorithm [11] that helps to ensure only permitted user can access particular patient's data at a certain time. As Fig. 1 illustrates, there are two data storage on the cloud namely Master Storage and Proxy Storage, both keep Electronic Patient Record (EPR) consisted of medical and health data. Both storages can be accessed through Data Access Service (DAS) containing REST service for the application client. Master storage keeps EPR that is encrypted with patient's public key so only personal record application on patient's device can decrypt it with his own private key. Proxy Storage keeps EPR as it is requested by monitoring application. EPR on Proxy storage is encrypted with public key of those who request and has permission to access, then it is signed by patient's private key for integrity and authenticity checks. Once doctor or medical staff retrieves EPR using monitoring application, they verify EPR using patient's public key to prove authenticity, before decrypting it with their own private key.

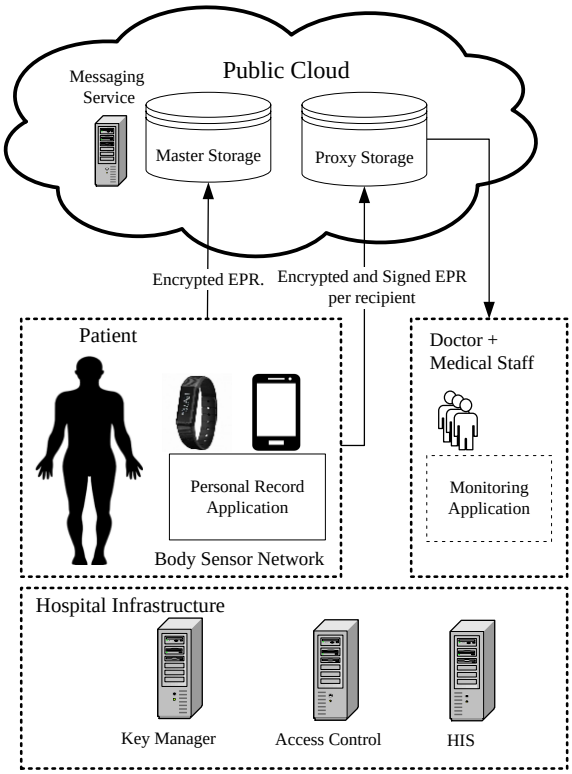

Fig. 1. Architecture overview.

Key Manager generates pair of keys, keep and provide public key for different user involved in the application system. *Access Control* contain policies that enable personal record application to validate who can access which patient's data at what level (e.g. cardiology doctor have write access their patients with heart disease, while nurses only have read access). Patient can manage access policy on his/her record therefore they take a full control of their own data. In the emergency situation, access policy will be override for medical ambulance staff for a period of time. Hospital Information Management (HIS) is integrated into our system in order to provide patient's medical record. Key Manager, Access Control and HIS are hosted on hospital infrastructure to minimize security risk. The messaging service on the cloud support sending notification when EPR is requested to access or when latest updated data is available on proxy storage.

IV. Security Management

The communication between users and cloud's server is over SSL, even though SSL ensure confidentiality and integrity of message exchanging between two parties. But public cloud-based server is considered untrusted as it is operated and maintained by the cloud provider company. The cloud provider has no legal right to access information belonging to the user. There is however a potential risk that the cloud-based server might be attacked by malicious program which may cause unauthorized data access. This section describes the detail implementation of security scheme that aims to protect data from both insider and outsider threats. The security scheme provides fine-grained access control over encrypted data on the cloud. Moreover, it also ensures integrity and authenticity of message transferred through cloud between patient and doctor. Throughout this paper we use $F(x, y, \ldots) \rightarrow z$ to denote the operation of running an algorithm F with inputs $x, y, \ldots$ and output z . Table I describes acronym used to explain security scheme in this paper.

TABLE I: Acronym Description

Acronym	Description
Prv^P, Pub^P	private key and public key of patient
Prv^R, Pub^R	private key and public key of doctor or medical staff
EPR^K	Encrypted E-Patient record with K as key

A. Storing Patient Record

Electronic Patient Record (EPR) is consisted of two parts: 1) health data which is collected from wearable device's sensors, 2) medical data which is recorded by medical staff and doctor on HIS. In order to use personal record application, a new patient needs to make registration on the application system. The following steps explain what happen in the background during registration.

1) *Key Manager* execute $KeyGen() \rightarrow (Prv^P, Pub^P)$ to generates key pair for patient using RSA algorithm.
2) Prv^P is securely stored on patient's smart phone device using AES asymmetry encryption algorithm to protect from unauthorized access.
3) EPR is loaded from HIS and encrypted with $Encrypt(EPR, Pub^P, S_n) \rightarrow EPR^{PubP}$ where S represent data attribute of vital sign and n is number of attributes to be encrypted,
4) EPR^{PubP} is saved on *Master Storage* through DAS.

B. Patient Record Retrieval

To protect privacy of patient information, EPR can be access by authorized entity through making an on-demand request using monitoring application. Monitoring application supports authorized parties to retrieve EPR by calling a

service on DAS. The following algorithm explains how DAS react with the access request.

```
UP^proxy = Query.EPR.Date(P_i , R_i)
UP^Master =Query.EPR.Date(P_i)
IF UP^proxy equals UP^Master
   THEN Response( EPR^PubR )
   ELSE
        Response( EPR^PubR )
      Update(P_i , R_i)
```

`Query.EPR.DATE()` denotes function to retrieve latest update date of EPR for given parameter, R_i is a particular requester and P_i is a particular patient.

`Response()` denotes function to response data back to the client.

`Update()` denotes function to refresh EPR from master storage to proxy storage for a particular requester.

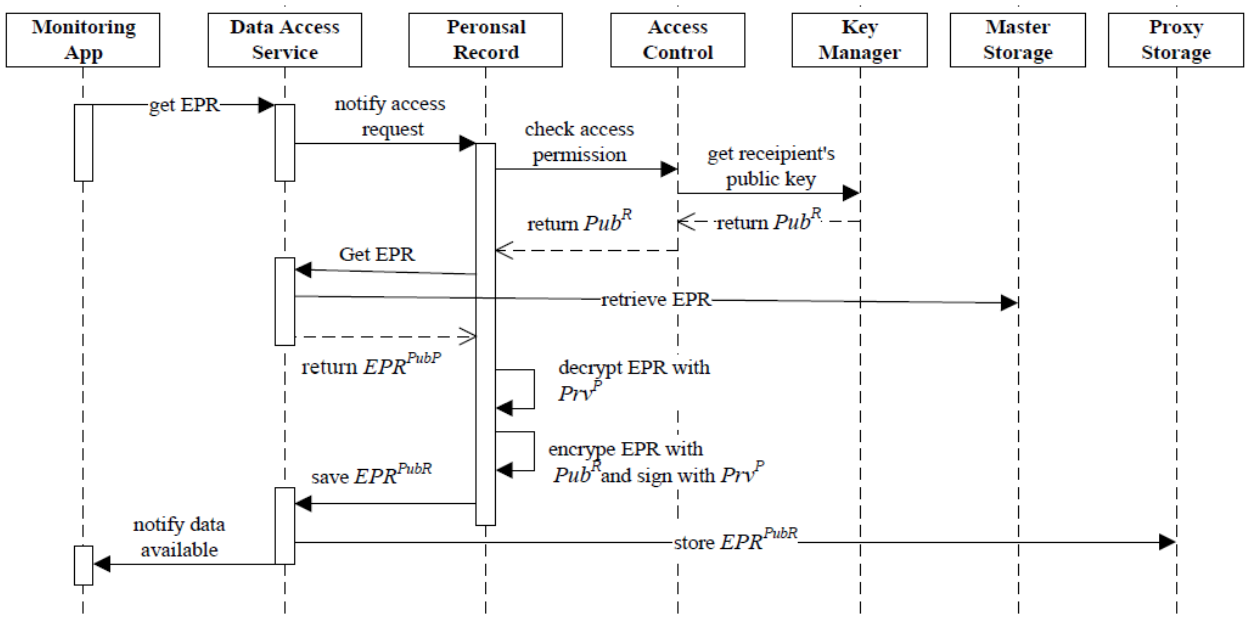

Fig. 2. Patient record updating workflow.

DAS determines whether it has up-to-date EPR. If up-to-date EPR is not found, DAS return latest EPR as it has and execute function `Update()`, which involves interaction between personal record application, master storage and components situated on hospital infrastructure, the flow of interaction is illustrated in Fig. 2. Firstly, DAS send notification about access request to patient's smart device through personal record application. After that, personal record application automatically checks with access control whether the requester has permission to access. Access control manages policies that are used to verify an access request. The access policy is a pair of identification number of patient and permitted user, with specified access level (e.g. write, read) and access type. There are two types of access: one-time and continuous access. One-time access allows requester to read EPR of a specific patient only once. When continuous permission access is granted, EPR is available anytime when it is requested until patient purge permission from the access policy. In order to access EPR from monitoring application, access privilege must be granted and record in list of access policy in advance. The monitoring application therefore has a function to facilitate this by generating QR code that contains requester's user identification number and allow patient's smart record to scan and grant access. After the permission is verified, patient's personal record application performs the following steps:

1) $Decrypt(EPR^{PubP}, Prv^P, S_n) \rightarrow EPR$ to decrypt EPR retrieved from master storage
2) $Encrypt(EPR, Pub^R, S_n) \rightarrow EPR^{PubR}$ to encrypt with requester's public key ,
3) $Sign(Prv^P, T) \rightarrow T^{PrvP}$ to sign a generate hash key denoted by T with patient's private key in order to ensure authenticity of EPR before sending together with EPR^{PubR} to proxy storage.

During emergency situation, proposed architecture also support user effectively. By analyzing collected data from sensors, personal record application can detect needs of emergency assistance, for example heart failure can be detected by high rate of heart pulse, blood pressure, and respiratory. The application automatically sends notification to medical staff for ambulance service, as well as giving one-time access privilege to medical staff that is responsible for the situation. Key manager generates pair of key which is used to protect confidential of patient's information.

V. Illustration

The proposed architecture has been implemented to monitor sampling group of patients at a local hospital in Chiang Rai, Thailand. Google App Engine (GAE) [12] is selected to be public cloud platform because of its competitive price as well as features such as auto scaling and easy deployment method. Master storage and proxy storage are implemented using GAE's Datastore which is a NoSQL document database designed to automatically scale to massive data sets, allowing applications to receive more traffic while maintaining a certain level of performance. This

suits very well with the requirement of master and proxy storage that have a high volume of data traffic. The architecture's software component is developed based on J2EE (Java 2 Enterprise Edition) [13], as Java is a programming language that can run on various platforms including GAE without rewriting source code specific to each platform, as well as its large ecosystem of open source tools . This gives us advantages when software system needs to be scaled, maintained and evolved. Two mobile application namely personal record and monitoring application are developed with Ionic as it is one of the most popular hybrid mobile application development frameworks with well-support developer community. The development can therefore take advantage of various open source plug-ins that facilitate the development and can be built to run on both Android and iOS.

A. Security Implementation

DAS is developed as collection of REST service API using Spring MVC framework, because the framework has simplified way to develop REST service through configuration and annotation in the source code. REST [14] service API provides service to manage patient record on master storage and proxy storage via the Hypertext Transfer Protocol over SSL (HTTPS). A REST service uses HTTP methods (GET, POST, PUT, DELETE, etc.) as well as unique URIs to access the underlying resources. Therefore, API is developed as combination of HTTP methods and URIs. The services exchange EPR as data object (including personal information and vital sign) with client as JSON (JavaScript Object Notation) format. Fig. 3 shows sample EPR in JSON. Encryptions are done on data attributes that represent vital sign such as pulse rate, respiratory rate, etc. so they are presented as cipher text in the data object. In order to ensure authenticity of message, digital signature is also included.

```
{
       patientId : 3234311243,
       recordDate : 01/11/2016,
       signature : 60891728367987549
       vitalSign : {
            pulseRate : 1213107243921127189732
            respiratoryRate : 9663173013977778
            systolic : 2628796167979706089
            diastolic : 5744549785445
            bodyTemperature : 87891279461234078
       }
  }
```

Fig. 3. Sample EPR in JSON.

B. Security Analysis

The confidentiality, integrity and authenticity of patient data are guarantees in our proposed architecture. RSA is public-private key cryptosystem that is well proved to resist unauthorized access. The master EPR is encrypted at patient's mobile device with his/her public key before sending and storing data on the cloud, therefore no other entities can access this data even though they have direct access to data storage on the cloud. When EPR need to be accessed by authorized entity, decryption of master EPR can only be done at patient's mobile device with his/her private key, before the data is encrypted with designated recipient's public key to ensure confidentiality. The security scheme included in our proposed architecture is resilient against man-in-the-middle attack [15] that the attacker attempt to alter data during communication between two entities in the system, because EPR is digital signed by patient's private key so it can be verify to ensure authenticity.

VI. Conclusion

In this paper, we address security challenge of mobile healthcare application that integrates to cloud-based storage. We propose an architecture that can overcome this challenge with security scheme that can protect data on the cloud from potential threats and guarantee confidentiality, integrity and authenticity of medical and healthcare data which is highly sensitive. The combination of encryption, decryption and interaction flow between different components helps to achieve this by using patient's smart phone as intermediary to secure their own personal data. The architecture has been implemented and has been pilot testing by sampling group of patients at a local hospital.

References

[1] V. Goyal, O. Pandey, A. Sahai, and B. Waters, "Attribute-based encryption for fine-grained access control of encrypted data," in *Proc. the 13th ACM Conference on Computer and Communications Security*, 2006.

[2] N. S. Kumar, G. R. Lakshmi, and B. Balamurugan, "Enhanced Attribute Based Encryption for Cloud Computing," in P*roc. the International Conference on Information and Communication Technologies*, Kochi, India, 2015.

[3] A. Lounis, A. Hadjidj, A. Bouabdallah, and Y. Challal, "Healing on the cloud: Secure cloud architecture for medical wireless sensor networks," *Future Generation Computer Systems*, vol. 55, pp. 266-277, 2016.

[4] M. Barua, X. Liang, R. Lu, and X. Shen, "ESPAC: Enabling security and patient-centric access control for ehealth in cloud computing," *International Journal of Security and Networks*, vol. 6, no. 2/3, pp. 67-76, 2011.

[5] F. Xhafa, J. Feng, Y. Zhang, X. Chen, and J. Li, "Privacy-aware attribute based PHR sharing with user accountability in cloud computing," *Journal of Supercomputing*, vol. 71, no. 5, pp. 1607–1619, 2015.

[6] A. Balu and K. Kuppusamy, "An expressive and provably secure cipher text policy attribute based encryption," *Information Sciences*, vol. 276, pp. 354-362, 2014.

[7] J. A. Akinyele, M. W. Pagano, M. D. Green, C. U. Lehmann, and Z. N. Peterson, "Securing electronic medical records using attribute based encryption on mobile devices," in *Proc. the 1st ACM Workshop on Security and Privacy in Smartphones and Mobile Devices*, 2011.

[8] A. Shamir, "Identity-based cryptosystems and signature schemes," *Advances in Cryptology. CRYPTO 1984 in Lecture Notes in Computer Science*, 1984.

[9] X. A. Wanga, J. Mab, F. Xhafa, M. Zhange, and X. Luoc, "Cost-effective secure E-health cloud system using identity based cryptographic techniques," *Future Generation Computer Systems*, vol. 67, pp. 245-254, 2017.

[10] J. A. Buchmann, E. Karatsiolis, and A. Wiesmaier, *Introduction to Public Key Infrastructure*, NY: Springer, 2013.

[11] S. Coutinho, *The Mathematics of Ciphers: Number Theory and RSA Cryptography*, MA: A K Peters, Ltd., 2013.

[12] Google Inc. Google cloud platform. [Online]. Available: https://cloud.google.com/appengine

[13] I. Singh, B. Steans, and M. Johnson, *Designing Enterprise Applications with the J2EE Platform*, New Jersey: Pearson Education, 2002.

[14] L. Richardson and M. Amundsen, *RESTful Web APIs*, CA: O'Reilly Media, 2013.

[15] C. Kaufman, R. Perlman, and M. Speciner, *Network Security: Private Communication in a Public World* (2nd Edition), NJ: Prentice Hall, 2002.

Nacha Chondamrongkul received the master of computer science in business consulting (M.Sc.) from Hochschule Furtwangen University, Germany in 2007. He has extensive experience working as a consultant on enterprise software implementation projects. Currently, he has been working as a lecturer at Software Engineering Department of Mae Fah Luang University where he has been taking on teaching and research responsibilities in the areas of software engineering. His research interest involves model-driven engineering, component-based software engineering, enterprise application development methodology, mobile computing and cloud computing.

Pattra Chondamrongkul received the master of business administration in logistic and supply chain management at mae fah luang university, thailand in 2013. She has been working as a lecturer at faculty of management science of chiang rai rajabhat university. She has been actively researching on multidiscipline areas such as logistics network, supply chain management, tourism management, digital economy, information technology, enterprise application and security.