Digital Certificate Management System for eHealth and mHealth Practitioners in Sri Lanka to Secure Medical Data

T.M.K.K. Jinasena[1#], R.G.N. Meegama[2] and R.B. Marasinghe[3]

[1,2]*Department of Computer Science, University of Sri Jayewardenepura*,
[3]*Department of Medical Education and Health, University of Sri Jayewardenepura,*
[#]kasunkosala@yahoo.com

Abstract—*eHealth and mHealth systems are getting more popular today; yet, vulnerabilities are much higher when the sensitive medical data being transferred through public networks. Therefore, it is essential to have a digital identification and authentication mechanism to authenticate peers in a digital world. Especially, it will help to avoid attacks such as man-in-the-middle attack. Although the digital certificates can solve this issue, it has not been used by the general public yet to protect their digital data. This is mainly due to their limited knowledge in IT and the complexity of the process. Thus, it is required to have a simple security tool to support encryption, digital signature, digital authentication, and integrity verification. However, we have developed a digital certificate management system to facilitate all these features including creating asymmetric key pairs, generating, signing, chaining and revoking certificates, and signing and verifying digital contents. Because it is a Java based application, it is platform independent; thus portable. In backend, it uses OpenSSL library. Moreover, it is capable of managing present RSA based certificates as well as the novel Elliptic Curve (EC) based certificates. Thus, it is more robust, future-proof and well-suited for mobile devices. However, a usability test was performed to evaluate its usability, efficiency and the effectiveness. 47 undergraduate and postgraduate students were voluntarily attended for the test and their responses were critically analysed. Compare to the conventional command line based method, 100% of user satisfaction has been gained by the developed tool. In conclusion, it is a simple, free and open source software for the public to secure their digital data.*

Keywords*— *Computer Security, Digital Certificate, PKI, eHealth & mHealth.

I. INTRODUCTION

eHealth and mHealth systems are getting more popular today; yet, vulnerabilities are much higher when the sensitive medical data being transferred through public networks. As the IBM survey in 2015, medical data has the highest per record security cost. Moreover, in USA, medical identity theft is the most rising crime today. All these emphasize the criticalness of security of medical data. However, identity cards play an important role in human society by providing a convenient way to prove the identity of a person to others. In the same way, digital certificates provide a way to prove the identity of a person or a device uniquely in the digital world. As more and more devices getting connected to public networks with the advancement of Mobile technology, Internet, and Internet of Things (IoT), it has become a crucial requirement to identify them and their activities uniquely. Especially, it is essential to avoid man-in-the-middle attacks. At present, digital certificates are being used to identify mainly servers, not the clients. This is because of the complexity of Issuing, distributing, renewing, revoking, and validating certificates. As a result neither servers nor the peers would be able to verify the real originator of the message. Thus, it would be not possible to implement security services such as non-repudiation. However, if one obtained a digital certificate, he/she has to pay a fee for the certificate. Therefore, the communities with thousands of user like universities, hospitals, government departments, agencies, etc. have to spend a huge amount of money for this annually. If an organization wants to identify its clients, it needs to have specialized people to manage those certificates. Besides, organizations can not have local authority on their identification system if they use the global verification chain as external parties can create valid certificates for them. Moreover, present encryption mechanisms are not suited to secure real-time video data in a mobile environment due to its time complexity. Therefore, it is necessary to have a simple tool to manage digital certificates within an organization especially in a non-technical environment like health. The objective of this research is to develop a simple but novel and robust digital certificate management system for eHealth and mHealth practitioners in Sri Lanka to manage their medical data securely (Abdalla et al., 2000; Al Nuaimi et al., n.d.; Andrews et al., n.d.; Armknecht et al., n.d.; Brands, n.d.; Gerard et al., n.d.; Gui-hong et al., n.d.; Dahlman, 2014; Zhou & al, 2007)

II. BACKGROUND

When a client visits a web site which has a certificate installed in the server, the web browser can verify the validity of that certificate through its certificate authorities. Especially, this is very helpful to avoid DNS attacks when you are visiting sites like email, internet banking or payment sites like PayPal. In fact, digital

certificates help you to identify any site before you enter your sensitive data such as passwords or pin numbers to that site or before you download software from it.

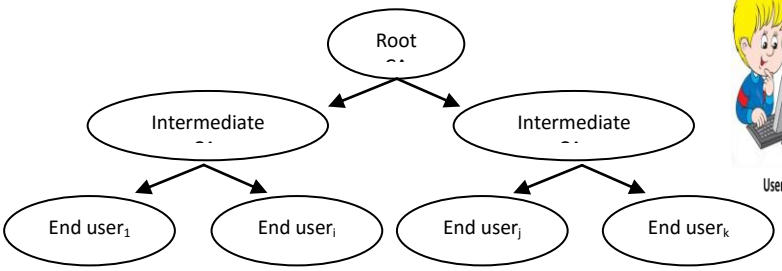

Figure 1: Certificate authority hierarchy

Thawte, GlobalSign, Combo, and DigiCert are some of the root certificate authorities who provide this service. However, there are number of intermediate certificate authorities in the world who are extending the service of those root certificate authorities. Practically it is not possible to installed signatures of all certificate authorities in a web browser. Thus, chain validation is being used. In chain validation, browser repeats the validation process till it gets the signature of a known trusted certificate authority (Nash et al., n.d.; Hunter et al., n.d.; Koehler, n.d.; Leavitt, n.d.). Figure 1 shows how the certificate authority hierarchy is organized.

III. METHODOLOGY

A free and open source library for cryptography named OpenSSL has been using in the backend to do the all cryptographic works. Thus it will automatically fix the bugs as you update the library. In order to make it platform independent, Java has been chosen as the development language. For the simplicity of the user, tasks have been divided into tabs so that user can focus on a single task at a time. Tabs have been organized in a way that it takes user from one step to the other as in wizards. Further, defaults values have been given to minimize the user inputs as well as give insight about the value that required for that field. Moreover, users can customize these values and save their own default values for later use.

Main steps of the process can be summarized as key pair generation, certificate sign request generation, signing a certificate, viewing a certificate, verifying a certificate, chaining certificate authorities, generating revoke list, and generating renew request. Apart from that, message digest generation, signing documents and verifying digital signatures can also be performed.

A. Creating a Digital Certificate

Figure 2 shows how the digital certificate is created by putting CA's signature for user's details and his/her public key.

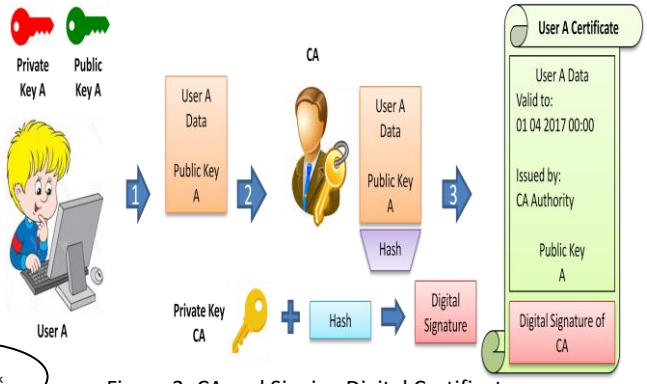

Figure 2: CA and Signing Digital Certificates

B. Generating a Digital Signature

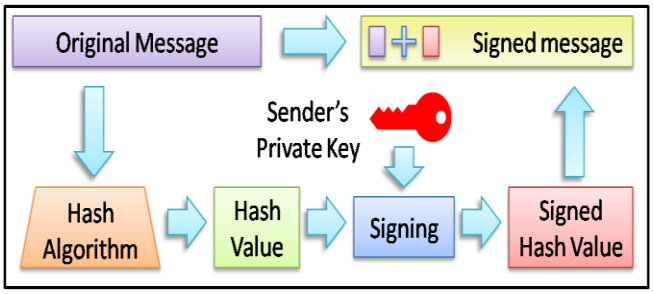

Figure 3: Signing a message

Figure 3 shows how the digital signature is generated using the message digest and the private key of the sender. This will provide a way for receiver to verify the integrity of the message and to authenticate the sender of the message.

C. Verifying the Integrity and Sender

Figure 4 shows how that verification process and authentication are happening.

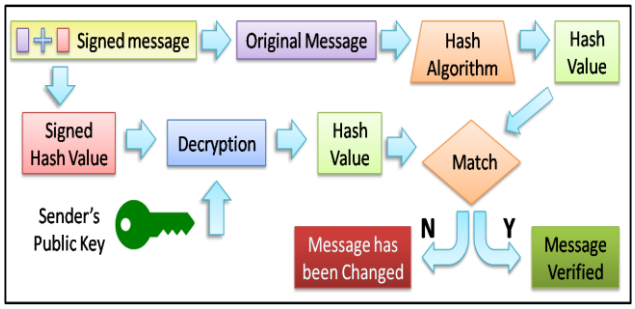

Figure 4: Verifying a Signed Message

Finally, a usability study was performed with the help of university students. First, they were given command line OpenSSL and a list of commands to execute. Secondly, they were asked to do the same task using the tool developed by this research. Thirdly, they have been asked to rate their experiences of both methods and finally, their feedbacks were critically analysis using statistical methods.

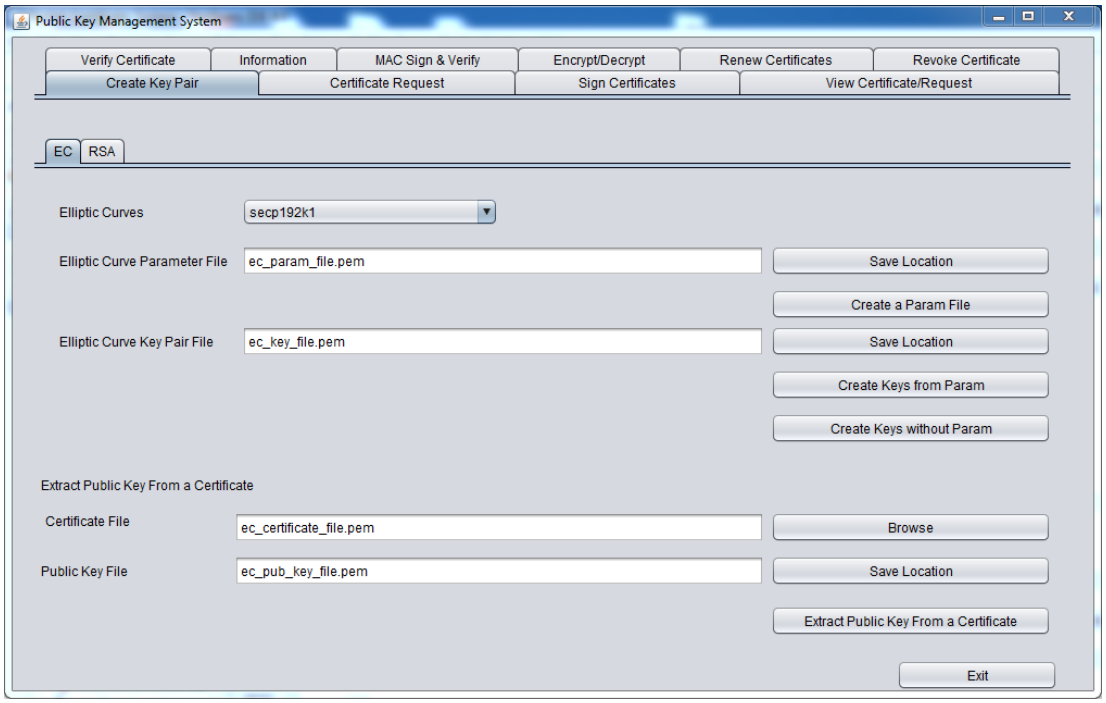

Figure 5: Key Generation

IV. RESULTS AND DISCUSSION

Present global public key infrastructure is based on RSA asymmetric cryptography system. However, in this research we have used a novel and robust cryptography method called Elliptic curve cryptography. Moreover, it is efficient than the present RSA method as it can provide the same security under 30 times less key size. Hence, it is suitable for mobile devices too. Elliptic curves are exponential. Besides, it is based on new mathematics on finite field rather than the integer factorisation. Thus, it is more robust than the present RSA method. Therefore, this public key infrastructure is more robust, future-proof and well-suited for mobile devices.

Figure 5 shows how the key pair and parameter files being created. **Error! Reference source not found.** shows how to sign a certificate using a root or an intermediate certificate. Besides, it allows you to create self sign certificates as well as root and intermediate certificates. Initially it was decided to carry out the usability test with the help of medical students by considering their higher IT knowledge compare to Doctors and other health workers. However, in the pre-test, it was found that most of the medical students were very uncomfortable with the command line although they have a relatively good knowledge in IT. Therefore, the computer science and IT undergraduate and post

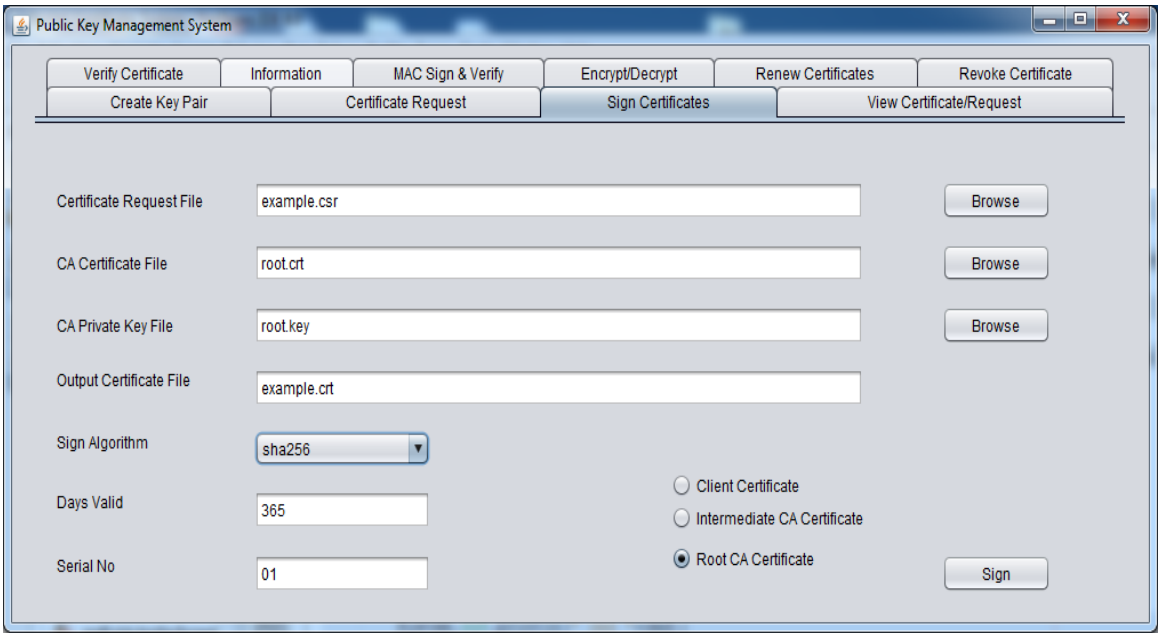

Figure 6: Signing Certificates

graduate students were chosen for the final test in order to compare user experiences of both. 47 students were voluntarily attended on the test. Most of them were familiar with the OpenSSL. Thus they were able to do all the command line tasks successfully with miner mistakes. Results show that 100% of the testers were voted this as a convenient, efficient, and less error prone tool for the public key management. In pre-test, some of the experts favour command line tool rather than this because it allows more freedom for customizing commands. However, the objective of this research is to empower general public with user friendly tool for public key cryptography and infrastructure management. Thus, the res

Works Cited

Abdalla, M., Shavitt, Y. & Wool, A., 2000. Key management for restricted multicast using broadcast encryption. *Networking, IEEE/ACM Transactions on*, 8(4), pp.443-54.

Al Nuaimi, N., AlShamsi, A., Mohamed, N. & Al-Jaroodi, J., n.d. e-Health cloud implementation issues and efforts. In *Industrial Engineering and Operations Management (*

Andrews, R.F., Williams, P. & Lin, J., n.d. Risk management for public key management infrastructure using digital certificates. Google Patents. 00128.

Armknecht, F. et al., n.d. An efficient implementation of trusted channels based on OpenSSL. In Proceedings of the 3rd

Brands, S.A., n.d. Rethinking public key infrastructures and digital certificates: building in privacy. Mit Press. 00768.

Dahlman, E.a.M.G.a.P.S.a.e.a., 2014. 5G wireless access: requirements and realization. IEEE Communications Magazine, 12(52), pp.42--47.

Gerard, P. et al., n.d. Cybersecurity in radiology: access of public hot spots and public Wi-Fi and prevention of cybercrimes and HIPAA violations. 201(6), pp.1186-89.

Gui-hong, L., Hua, Z. & Gui-zhi, L., n.d. Building a secure web server based on OpenSSL and apache. In E-Business and E-Government (

Hunter, T.B., Weinstein, R.S. & Krupinski, E.A., n.d. State medical licensure for telemedicine and teleradiology. 21(4), pp.315-18.

Koehler, S.C., n.d. Method and system for authenticating digital certificates issued by an authentication hierarchy. Google Patents. 00149.

Leavitt, N., n.d. Internet security under attack: The undermining of digital certificates. 44(12), pp.17-20. 00052.

Nash, A., Duane, W. & Joseph, C., n.d. PKI: Implementing and Managing E-security.

Zhou, Y. & al, e., 2007. Access control in wireless sensor networks. Ad Hoc Networks, 5(1), pp.3-13.