34th Annual International Conference of the IEEE EMBS
San Diego, California USA, 28 August - 1 September, 2012

Device Interoperability and Authentication for Telemedical Appliance based on the ISO/IEEE 11073 Personal Health Device (PHD) Standards*

Luther Paul R. Caranguian, Susan Pancho-Festin, Ph.D., Luis G. Sison, Ph.D.

Abstract— **In this study, we focused on the interoperability and authentication of medical devices in the context of telemedical systems. A recent standard called the ISO/IEEE 11073 Personal Health Device (X73-PHD) Standards addresses the device interoperability problem by defining common protocols for agent (medical device) and manager (appliance) interface. The X73-PHD standard however has not addressed security and authentication of medical devices which is important in establishing integrity of a telemedical system. We have designed and implemented a security policy within the X73-PHD standards. The policy will enable device authentication using Asymmetric-Key Cryptography and the RSA algorithm as the digital signature scheme. We used two approaches for performing the digital signatures: direct software implementation and use of embedded security modules (ESM). The two approaches were evaluated and compared in terms of execution time and memory requirement. For the standard 2048-bit RSA, ESM calculates digital signatures only 12% of the total time for the direct implementation. Moreover, analysis shows that ESM offers more security advantage such as secure storage of keys compared to using direct implementation. Interoperability with other systems was verified by testing the system with LNI Healthlink, a manager software that implements the X73-PHD standard. Lastly, security analysis was done and the system's response to common attacks on authentication systems was analyzed and several measures were implemented to protect the system against them.**

I. Introduction

In a telemedical system, medical services can be delivered to patients in remote locations. A general telemedical system consists of the central authority, usually a domain expert or hospital which communicates to a telemedical appliance that is located remotely. The biomedical signals from the medical devices along with patient information are sent by the telemedical appliance to a domain expert for analysis and diagnosis.

Recent studies in telemedicine involve development of standards for interoperability. Interoperability is the capability of several systems and devices to cooperate using exchanged information regardless of type, model and manufacturer Standards such as HL7 and ISO/IEEE 11073 were developed, implemented and evaluated in many studies [1].

Security and authentication are also crucial concerns in a telemedical system because the information is private, sensitive and critical [2]. Standards within ISO/TC 215 (Working Group 4) have defined security standards for health informatics [3].

Securing a telemedicine system is important because medical services are critical to patient's health and life. A telemedical appliance which is deployed in a remote location needs to be authenticated by a central authority to assure data integrity and valid identity. At the same time, authentication of the medical devices and instruments connected to the telemedical appliance is necessary for anti-counterfeiting.

A number of standards and studies were already implemented to address authentication of the telemedical appliance by a central authority. However, no standards or studies were made for authenticating medical devices. Morever, in implementing a security mechanism the ISO/IEEE 11073-PHD standard [1] for interfacing medical devices must be considered. This will facilitate device interoperability and easier adoption of device manufacturers.

A. Objectives

In this study, we develop a security protocol within the ISO/IEEE 11073-PHD standard to support medical device authentication. Several mechanisms will be tested and evaluated. To achieve this, the following objectives must be met.

- Implement and evaluate the ISO/IEEE 11073-PHD standard on a candidate telemedical appliance
- Design and implement a security protocol in authenticating the medical devices
- Implement security protocol in medical device for two scenarios:
  1. generic microprocessor-based system
  2. microprocessor-based system with Embedded Security Module (ESM)

II. Related Work

The ISO/IEEE 11073 Personal Health Device (X73-PHD) Standards addresses the device interoperability problem by defining common protocols for agent (medical device) and manager (appliance) interface. The standard is

*Research funded by the Department of Science and Technology-Science Education Institute (DOST-SEI) and the Engineering Research and Development for Technology (ERDT)

L. R Caranguian is a graduate student in the MS Electrical Engineering (Instrumentation and Control) program of University of the Philippines Diliman. (phone: +63-917-527-5732; e-mail: ldcaranguian@up.edu.ph).

S. P. Festin is with the Department of Computer Science, University of the Philippines Diliman. (e-mail: susan.pancho@up.edu.ph).

L. G. Sison is with the Electrical and Electronics Engineering Institute, University of the Philippines Diliman. (e-mail: sison@eee.upd.edu.ph).

978-1-4577-1787-1/12/$26.00 ©2012 IEEE

expected to be well adopted by the market because of the promotion by the Continua Health Alliance, an organization with over 250 member companies involved in telemedical initiatives [1]. The standard is gaining popularity and several papers describing implementation of the standard are already published [4], [5].

Security and authentication is important concern in a telemedical system because we are dealing with private, sensitive and critical information. A number of studies has been done to secure and provide authentication of a remote telemedical system by a central authority [2]. However, no studies were found to address authentication of the medical devices attached to the telemedical applicance. This is an equally important concern especially with the gaining popularity of an open standard such as the X73-PHD, we would eventually face the problem of device counterfeiting.

Even non-medical devices such as USB drives, batteries and printer cartridges have started implementing device authentication to address counterfeiting issues. A relevant standard is the IEEE Standard 1667-2009 [6] which is the authentication in host attachments of transient storage devices (TSD's). We use the standard as a guideline in designing an authentication policy for medical devices.

III. Methodology

A. X73-PHD Standard Implementation

The manager is implemented in a Linux-based PC using C++ as programming language. Figure 1 shows the general architecture for the manager's software. The Physical and Transport layers are covered by the USB 2.0 Specification [7] and USB PHDC Class Definitions [8]. We have used LibUSB [9] (an open-source library in Unix-like systems) for the USB communication.

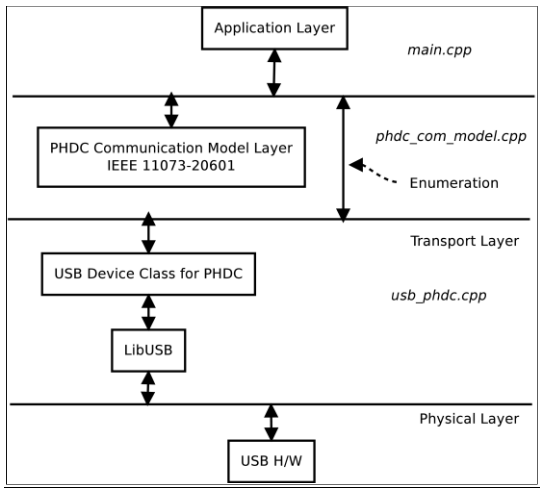

Figure 1. Linux-based Manager Architecture

The interface between the transport layer and the application layer is the communication model defined by the x73-PHD standard. We have used the state machine diagram and several examples from the x73-PHD standard [1] as a starting point in code development for the PHDC Communication Model Layer. The application layer is a routine to demonstrate the agent-manager interaction.

We developed a reference agent (pulse-oximeter device) using Freescale DemoJM development kit interfaced with a pulse-oximeter module. The DemoJM gathers actual readings from patient through UART communication with Envitech ChipOx development module. Figure 2 shows the block diagram of the agent implementation.

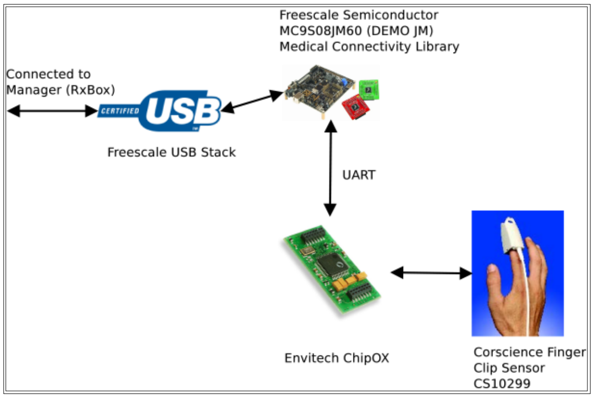

Figure 2. Agent Device Implementation Block Diagram

We performed end-to-end testing using the Linux-based manager and the pulse oximeter. We have verified interoperability and compliance of the manager by testing it with the demo program of the Medical Connectivity Library of the Freescale DemoJM board. The pulse oximeter agent was also verified using the Healthlink software from LNI. The software implements the x73-PHD standard for manager and runs on windows PC and is certified by the Continua Health Alliance.

B. Authentication Policy and Security Model

We designed an authentication policy using Asymmetric Cryptography which is incorporated in the X73-PHD standard. Figure 3 shows the modified association

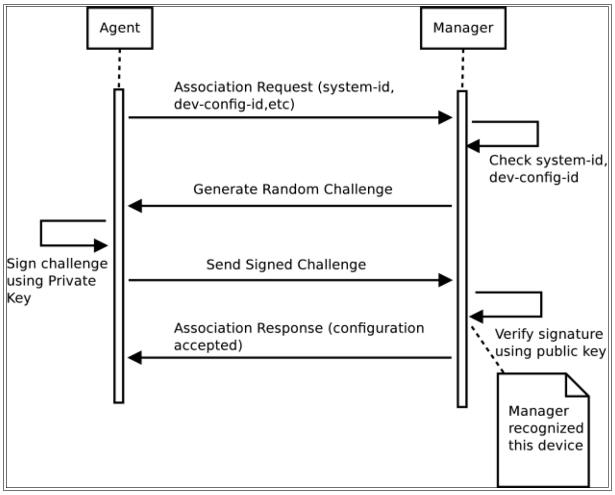

Figure 3. Modified Association Request of Agent to Manager

communication diagram. After the association request, the manager challenges the agent with a random data. The agent will sign the data using its private key and sends back the signature. The manager verifies the signature using the public key. It will associate itself with the agent when the verification is successful.

RSA algorithm was chosen for this study because of its popularity and because it enables us to benchmark our implementation with other authentication systems. We have used Crypto++ and MatrixSSL open-source libraries for the RSA implementation in manager and agent respectively.

We implemented an alternative security design in which we use an external embedded security module (ESM). We have used the Atmel Cryptographic Controller as the ESM. The Atmel ESM Trusted Platform Module (TPM) development kit *i.e.* it follows the TPM standards [TPM] in performing cryptographic processes. Figure 4 shows the block diagram when using the Atmel ESM together with the whole agent device.

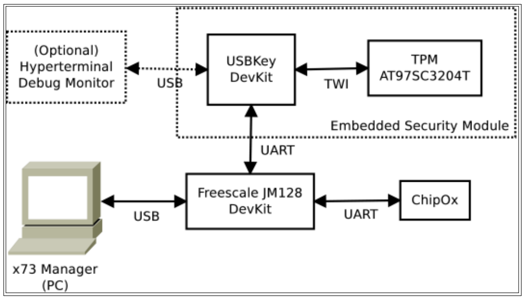

Figure 4. Atmel ESM and DemoJM Interface

IV. Results and Analysis

A. Performance Testing

We tested the performance of the RSA implementation in terms of execution time and memory requirements. We characterized the performance of the authentication system throughout different key sizes. We compared two different implementations of RSA using the MatrixSSL library and DemoJM microcontroller with the Atmel Cryptographic Controller. The two implementations using the MatrixSSL library are compared: (1) default implementation using Montgomery Reduction Algorithm and (2) optimized implementation using Chinese Remainder Theorem.

Figure 5 shows the graph of the testing results in log-log scale. Results show that on most cases, the TPM performs better than the direct implementation. At less than 512-bit keysize the Chinese Remainder Theorem implementation performs better than the TPM. However, the TPM as a much lower rate of increase in execution time compared to the microcontroller implementation. This is an important property of the TPM because we are able to scale keysize (thus increasing level of security) with smaller effect on the execution time. As an illustration, if we look at the 2048-bit and 3072-bit data, the microcontroller implementation increases execution time by almost 350ms or a little less than 6 minutes while the TPM increases execution time by less than 2 seconds.

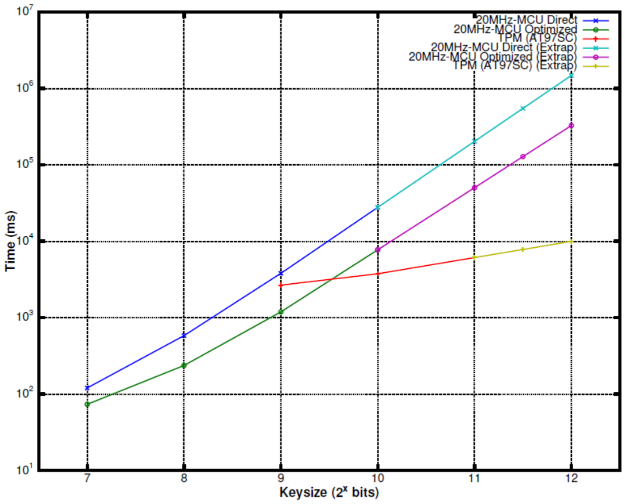

Figure 5. Average Execution Time for Test Cases

Table 1 shows the recommended keysize from RSA laboratories. Currently, the recommendation is to use 2048 bits for authentication. From the results of testing, we see that using direct implementation and optimized algorithm takes 203s (3.38 minutes) and 50.3s (0.84 minutes), respectively. This is unacceptable performance since the authentication protocol includes random authentication requests during operating state. Halting the device for one minute just to perform authentication is not acceptable especially for medical devices. The TPM takes only 6.12 seconds to perform authentication. This is a much better performance but may be unacceptable for devices with real-time data reporting such as ECG.

Table 1. Recommended Keysize for RSA Algorithm

Year	Until 2010	**2010-2030**	Above 2030
Keysize	1024 bits	**2048 bits**	3072 bits

Memory constrains possess another issue for the microcontroller implementation. The RSA calculations requires significant memory resources for its `malloc()` calls. We have calculated that we only have 7.875KB of SRAM left for `malloc()` calls out of the 16KB available for JM128. Implementing 2048-bit RSA using microcontroller might work for the direct implementation since it requires 7.17KB memory for calculation. However, for the optimized algorithm, it requires 11.6KB of memory which is more than what is available. When we opt to use TPM, we would not have this constraint because all calculations are done in the TPM.

B. Security Analysis

We performed security analysis of the authentication system based on the paper of Bau and Mitchel [10]. They described a uniform approach in evaluation of security systems and suggested analysis approach by determining the security model. The following three points summarizes the security model.

System Model: system behavior for intended and unintended inputs and operating conditions

Threat Model: attacker's behavior, computational resources and access to system

Security Properties: what we hope to prevent the attacker from violating

According to Bau and Mitchel, a system is "secure if the system design achieves the desired security properties against the chosen threat model." Their definition of system security is dependent on how extensive our knowledge of the system model, threat model and security properties.

We define the threat model by identifying common attacks on authentication systems from literature. We then identify if the current security properties are sufficient against common attacks and propose improvements when necessary. The following list enumerates common attacks along with the analysis of our systems security properties.

Dictionary/Replay Attacks: Dictionary or replay attacks are done by storing possible challenge data and their corresponding signatures into databases. For our system to prevent this threat, we must include additional information in the challenge data to insure that challenges are fresh and not replayed. We have added a timestamp and a session ID on the challenge packet to protect the system against dictionary and replay attacks.

Attacks on RSA Algorithm: Although the RSA is widely used, it has some weaknesses and there are feasible attacks to break the algorithm. A possible attack is to factor the public key to get the private key. This is easily exploited when the keysize is not large enough, and common modulus and low exponent are used. We have used the recommended 2048-bit keysize in the system and followed rules in generating keysize from the TPM standard to protect the system from exploiting the RSA algorithm.

Agent-Manager Collaboration: This attack happens given a scenario where the agent device is a counterfeit and the manager is hacked to bypass authentication. This would allow any counterfeit device to be associated with the manager and capture measurements from the user or patient. We propose to authenticate reports of the manager using signature generated through the agent before sending to the central authority or the hospital. A hash file of the whole report must be signed by the agent and then included when sending to the central authority. This method would verify that the report is not altered and the agent is genuine.

Compromising Key Distribution: In any authentication system, key handling and distribution is very crucial. When the private keys are compromised, the whole authentication system would become useless. It is important to make sure that the private key should never be computed using the public key or any other public information. The TPM has defined key hierarchies to protect the system against compromising the keys. The TPM is also not allowed to transfer keys without encryption and is storing keys in a secure memory in the IC.

V. CONCLUSION AND RECOMMENDATIONS

In summary, we have implemented a manager system using a Linux-based PC and a pulse oximeter agent device using Freescale JM128 microcontroller development kit. The implementations follow the ISO/IEEE 11073-PHD Standard and use Universal Serial Bus (USB) as communication protocol. We have performed end-to-end testing and interoperability test with existing manager software which also implements the ISO/IEEE 11073-PHD Standard.

We were able to design an authentication protocol within the standard to address the problem of device counterfeiting. Using existing protocols and methods in Public Key Infrastructure (PKI) and Asymmetric Cryptography, we can be able to determine the authenticity of the agent device automatically.

It is recommended to present the results of this study to the ISO/IEEE working committee for x73-PHD standard. The authentication protocol design would need further peer review especially from the people involved with the x73-PHD standard. It is recommended that the authentication protocol would be included in the standard or would be defined as a separate standard so that the industry would adapt it. It is also recommended that the demonstration system developed in this study would undergo the formal interoperability and conformance test done by Continua Health Alliance. This testing requires a sizable fee, detailed documentation of the implementation and actual physical test of the device in their test centers.

REFERENCES

[1] "ISO/IEEE 11073-20601-2008: Health Informatics-Personal Health Device Communication Part 20601: Application Profile- Optimized Exchange Protocol," Dec. 2008.

[2] Stapic, Z., Vrcek, N., and Hajdin, G., "Evaluation of security and privacy issues in integrated mobile Telemedical system," in *Information Technology Interfaces, 2008. ITI 2008. 30th International Conference on*, ISSN 1330-1012, Jun. 2008.

[3] "ISO/TR 11633: Health Informatics -- Information security management," 2009.

[4] Trigo, J.D., Chiarugi, F., Alesanco, A., Martinez-Espronceda, M., Chronaki, C.E., Escayola, J., Martinez, I., and Garcia, J., "Standard-compliant real-time transmission of ECGs: Harmonization of ISO/IEEE 11073-PHD and SCP-ECG," in *Engineering in Medicine and Biology Society*, 2009, ISSN 1557-170X, Sep 2009.

[5] Altuna, Ander, Lopez, Manrique, Carot-Nemesio, Santiago, and de las Heras, Pedro, "OPENHEALTH ASSISTANT: The OpenHealth FLOSS Implementation of the ISO/IEEE 11073-20601 Standard." in *AALIANCE conference - Malaga, Spain*, Mar 2010.

[6] "IEEE Std 1667-2009, IEEE Standard for Authentication in Host Attachments of Transient Storage Devices," 2009.

[7] "Universal Serial Bus Specification Revision 2.0," April 2000.

[8] "USB Device Class Definition for Personal Healthcare Devices," 2007.

[9] "Libusb-1.0 API Reference," April 2010.

[10] Bau, Jason and Mitchell, John C., "Security Modeling and Analysis," in *Security Privacy IEEE*, Volume 9, pp. 18: 25, May 2011.