Asian Journal of Computer and Information Systems (ISSN: 2321 – 5658)
Volume 02 – Issue 02, April 2014

Developing a Secure Messaging Platform in Mobile SMS Using Advanced Encryption Standard Concept

[1] Veronicah Mutua, [2]Andrew Mwaura

University of Nairobi
Nairobi, Kenya
[1] *Vmutua {at} uonbi.ac.ke*
[2] *Andrew.mwaura {at} uonbi.ac.ke*

ABSTRACT— *Billions of text messages are sent daily through GSM networks sent in plain text format which makes them susceptible to various attacks. With the increasing number of software crackers available for free in the internet, studies have shown SMS continues to suffer from several security vulnerabilities and loopholes. The aim of this project was to develop a secure messaging platform for Java enabled phones to securely send encrypted short messages via the GSM network.. The application was developed in MDIP 2.0 and CDLC 1.1configurations and applied symmetric encryption concepts. The experimental results revealed that the mobile application was a ble to encrypt, decrypt, send and receive text messages without adding changing the size of the packets.*

Keywords— Connected Limited Device Configuration (CLDC), Global System for Mobile Communication (GSM), Mobile Information Device Profile (MIDP), Short Messaging Services (SMS).

1. INTRODUCTION

The emergence of Short Messaging services (SMS) has extensively transformed the nature of communication and information sharing [1]. Billions of text messages are sent daily in the world sent in plain text format and hence user privacy and security is not assured. With the increasing number of software crackers available for free in the internet, SMS continues to suffer from security vulnerabilities and loopholes[8]. Wireless networks and mobile (ad hoc) networks have become vulnerable to various malicious attacks especially with the existence of software cracking tools such as 'juju'(a software tool that can sniff record and spy on text messages and voice)[9]. Studies like the one carried out by PC world, show that smart phones and to a larger extent mobile devices are more vulnerable to attacks because consumers of these devices are oblivious of the eminent threats that faces them[10]. MocanaGroup also demonstrate that 'Mobile devices have become critical business, military and industrial production tools, carrying valuable data well worth destroying, corrupting and, most importantly, stealing' [1].

With the current need for privacy of SMS, there is a universal need for solutions that can mitigate against threats in mobile communication. This project aimed at developing a secure mobile messaging application (in Java) that would allow users to send encrypted messages securely via the GSM network. The project targeted java enabled feature phones because of the high numbers of users in Kenya. The application was developed in MDIP 2.0 and CDLC 1.1configurations and applied symmetric encryption concepts.

Problem statement

The existing short messaging (SMS) platform allows texts to be sent in plain text hence the SMS can easily be intercepted and replayed using the existing software crackers. In this age where privacy is highly valued, it's expedient to come develop a s ecure messaging application that will ensure confidentiality, integrity and the availability of SMS sent via the GSM network.

2. TECHNOLOGICAL TRENDS IN MOBILE SMS

The emergence of mobile technology services in the recent years extensively transformed mobile communication and information sharing. Among the most popular services is the Short Message services (SMS) in which billions of SMS's are sent daily across the world [2]. Short messages service has not only been used for individual conversations but in corporate(mobil e banking), social and the political world[3].On the other hand, Studies have shown that the SMS channel is vulnerable for the man in the middle attacks and other hacking attacks[1]. With billions of text messages sent in plain text, the integrity of the t ext messages and the privacy of the users is bound to be breached especially now with the availability of many software crackers available for free in the inter net[4]. Several studies have been carried out with the aim of improving the SMS platform[9][3]. At the same time, several studies have also shown the need for using encryption in the SMS platform [3].

3. METHODOLOGY

Choosing the appropriate methodology for software projects always plays a huge role in determining the success of software product [5]. In the development of secure messaging mobile encryption application, various factors were put into consideration before settling on objectory use case approach that was proposed by L. Jacobson in 1994. The process involved identification of functional requirement from which use case artifacts were developed, after which, the dynamic and static behavior of the system were analyzed and modeled. The modeling of static behaviors was done through identification of objects and classes which were represented using unified modeling language (UML) diagrams. The dynamic aspects of the system were modeled using sequence, interactive, state diagrams and collaboration diagrams. The project adopted Extreme Programming software methodology because it was the most appropriate software development method for the production of a secure messaging mobile app [6].

3.1 The architecture of the proposed system

The mobile application a multi-layered application consisting of user experience, business and data layers. The application was developed is a rich client .The business and data layer services are located on the device itself. The presentation layers contains the user interface components(messaging menu) by which the user is able to compose the messages, read inbox and interact with other items. The business layer holds the encryption and decryption logic. It also defines the application facades and other customizable components that can be used for integrating with existing mobile banking. On the other hand, the data layer is responsible for storing incoming messages and making the messages available through the cache. The figure 2 below shows the secure messaging mobile application architecture with components grouped by the areas of concern.

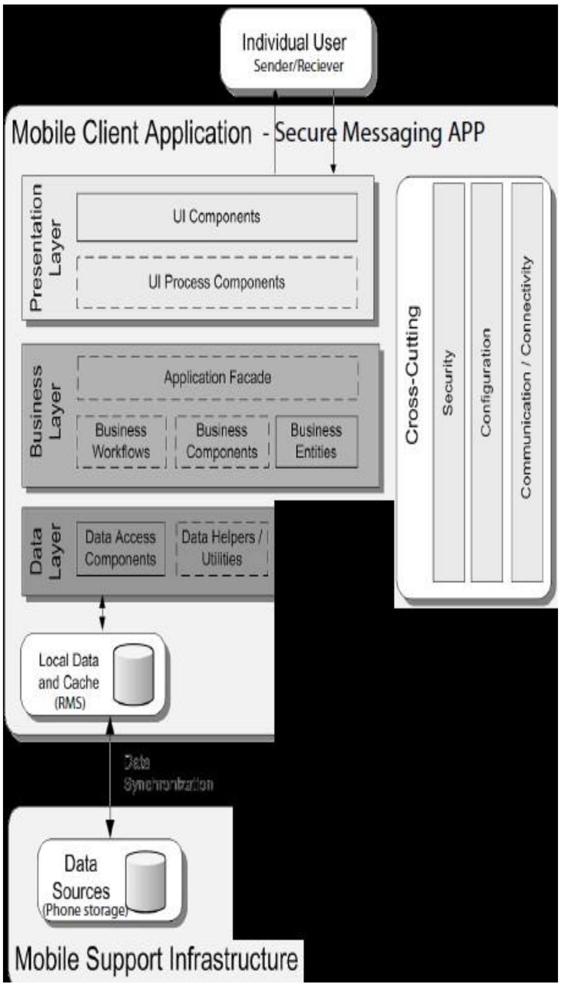


Figure 2: The proposed system architecture

3.2 SMS architecture in relation to secure messaging mobile application.

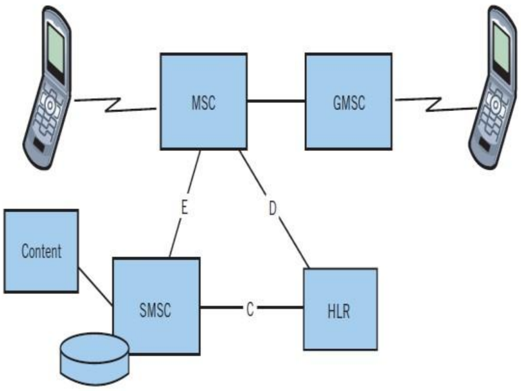


Fig.1 SMS architecure

The secure messaging application will run on both the sender's and the receiver's handsets. The users will be able to encrypt the Message at his/her handset using the application. The system will provide the users with a similar but different messaging menu for sending and receiving secure messages. The encrypted SMS is sent to the MSC which then forwards the message to the SMC in an encrypted format. The SMSC then looks at the recipient's number using the HLR, it then forwards the encrypted text to the nearby MSC which forwards the encrypted message for the recipient.

4. RESULTS AND EVALUATION

The secure messaging application was developed using Java Micro-edition for small devices(J2ME). The version 1.0 of the application targeted users with low end java enabled phones owing to the fact that large number of users (42%) in Kenya own Java enabled phones. Netbeans 7.0 was used as the development[7] environment with JDK 3.0 emulators for testing the application. Installation on the phone(Nokia 6030) was done by transferring jar and jad files to the phone via Bluetooth. Other phones(So ny Erickson, China made phones and LG phones) were used for testing and the results are shown below.

4.1 Results

The experimental results were carried out by installing the system on various mobile devices models. The messages were then sent and the results deduced.

4.1.1 Compatibility level

The application was installed on various phones to test its compatibility level and the user interface (UI). The secure messaging was developed in CLDC 1.1 and MIDP 2.0. These settings therefore played a role in determining the UI when installed on other phones. Other optional configuration settings that determined installation are:

- Advanced multimedia supplements API 1.0
- Java Bluetooth API for mobile devices
- Wireless messaging API 2.0
- Security and Trust services API for J2ME
- Content Handler 1.1

The following results were concluded after successful installation of the secure messaging application as shown below.

Phone model	**OS/Java capabilities**	**compatibility**
Mobile Emulator JDK 3.0 devices	yes	Successful Installation
Nokia 6030	yes	Successful Installation. The application adopts Phone's UI
ZTel(China model)	yes	Successful Installation. The application adopts Phone's UI
Sony Erickson	Yes	Failed Installation. Java not compatible with the MDIP CDLC 2.1
Huwawei Safaricom phone('Kabambe')	Yes	Successful Installation. The application adopts Phone's UI

Table 1.0 compatibility levels

4.2 Sample plain-text SMS, encrypted SMS, deciphered SMS

The system was tested to determine whether the encryption and decryption functional were working effectively. Table 2.0 below shows the various system results

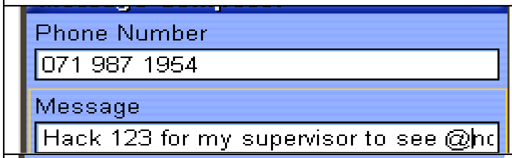
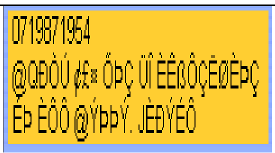
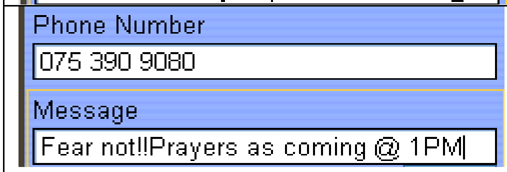
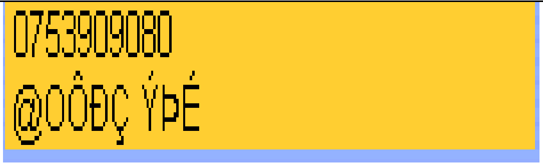
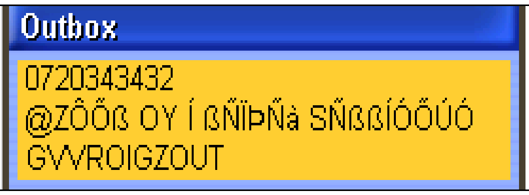
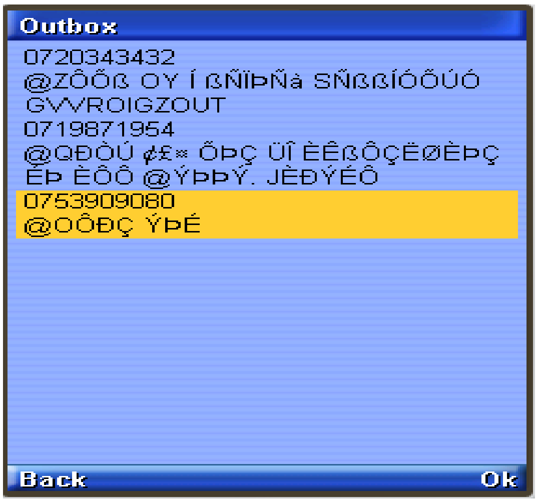
Plain text	**Encrypted text**
 Phone Number 071 987 1954 Message Hack 123 for my supervisor to see @hc	 0719871954 @QĐÒÚ ¢£¤ ŐÞÇ ÜÎ ÈÊßÔÇËØÈÞÇ ÉÞ ÈÔÔ @ÝÞÞÝ. JÈĐÝÉÔ
 Phone Number 075 390 9080 Message Fear not!!Prayers as coming @ 1PM	 0753909080 @OÔĐÇ ÝÞÉ
0720343432 This IS a secret messaging Application.	 Outbox 0720343432 @ZÔŐß OY Í ßÑÏÞÑà SÑßßÍÓŐÚÓ GVVROIGZOUT

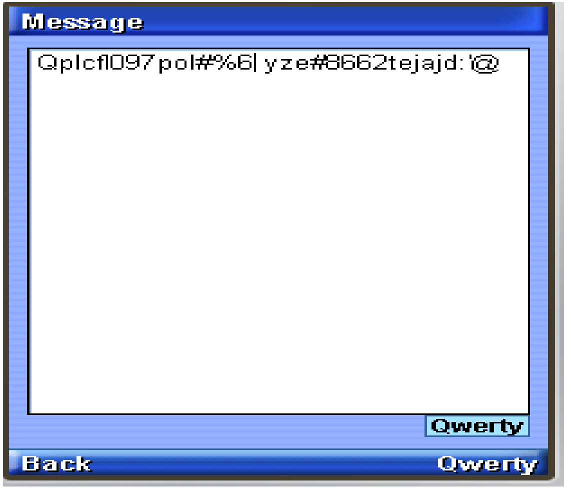
Table 2.0 system results



0720343432
@ZÔŐß OY Í ßÑÏÞÑà SÑßßÍÓŐÚÓ
GVVROIGZOUT
0719871954
@QĐÒÚ ¢£¤ ŐÞÇ ÜÎ ÈÊßÔÇËØÈÞÇ
ÉÞ ÈÔÔ @ÝÞÞÝ. JÈĐÝÉÔ
0753909080
@OÔĐÇ ÝÞÉ

Figure 3: outbox summary

Using a wrong key also decrypts the text messages further as shown below.



Qplcfl097pol#%6| yze#8662tejajd:'@

Figure 4: Encrypt futher

4.3 Analysis of System Results

The system results was observed in terms of compatibility with various phones, the ability to encrypt and decrypt text messag es without adding the size of the SMS, the time interval for sending and receiving messages and finally, the robustness of the secure messaging application.

5. DISCUSSION

The existing SMS architecture is prone to sniffing and replay attacks[8].With increasing number of hacking tools such as juju and Smart Sniff, its' evident that security in SMS architecture needs to be enhanced. It is known that SMS travels as plain text and privacy of the contents of SMS cannot be guaranteed, not only over the air, but also when such messages are stored on the handset.This project achieved its main objective by developing a secure SMS tool that enhances SMS security and data integrity of SMS. The application was tested using various Java enabled phones. It's however worth to note that , not all Java enabled phones were compatible with the application. The phone requirements included MDIP 2.0 or MDIP 2.2 and CDLC 1.0 and above. Most Chinese made mobile devices and Nokia have these features. However, the researcher would recommend Nokia 3500, 6030, 6200 series among other Java enabled phones because of ease of installation and operation.

5.6 Conclusion

The project ensured the development of a secure messaging application that targeted Java enabled phones. However, with the rising number of smart phones, users with high end phones may find this application incompatible with their mobile devices. The application adopted symmetric encryption concepts where the same key is used to decrypt and encrypt messages. Therefore, Key distribution remains a challenge as the sender and the receiver must know the keys prior to exchanging of messages.

Although the application gives room for the changing of the Keys, in an event that an attacker identifies the secret keys, he /she can change the keys without the knowledge of the users hence a limitation to the system. Therefore, future research and development needs to be carried to ensure that asymmetric version of the application are developed.

In conclusion, the outcome of the system evaluation showed that the system could send and receive messages securely encrypted messages. Experimental results showed that the system has the capacity to decrypt and encrypt messages without adding the si ze of the packet. Unlike the normal SMS that are sent in plain text via GSM network, messages sent via secure messaging APP remained encrypted during the transit.

Future works can be focused on improving the application to accommodate asymmetric encryption systems and improve compatibility to other versions of mobile devices and operating systems.

6. REFERENCES

[1]. Mocana(2012).*Mobile Security*available at : http://www.mocana.com,Retrieved 23rd September 2012

[2]. Baron, S., Patterson, A., & Harris, K. (2006): Beyond technology acceptance: *understandingconsumer practice.* International Journal of Service Industry Management. Vol. 17 No.2,2006. pp.111-135. Emerald Group Publishing Limited.

[3]. Herzberg, A. 2003. Payments and banking with mobile personal sevices.*Communications of the ACM* .Volume 46, Issue 5 (May 2003) Wireless networking security Pages: 53 58 ISSN: 0001-0782

[4]. R. El-Khalil, A. D. Keromytis(2007). *Hydan: Hiding Information in Program Binaries*, The 9th International Conference on Information and Communications Security (ICICS 2007),Zhengzhou, China, 2007

[5]. Beck, K. (1999a). Embracing Change With Extreme Programming. IEEE Computer 32(10): 70-77.

[6]. Beck, K. (1999b). Extreme programming explained: Embrace change. Reading, Mass., Addison-Wesley.

[7]. Njenga, A. D. K.(2009). Mobile phone banking: Usage experiences in Kenya. <http://www.strathmore.edu/pdf/ictc-08/mobile-banking.pdf > Accessed on 30th August, 2012.

[8]. MobiInfo(2012). *mobile information trends* Retrieved 23rdSeptember 2012available at : http://www.mobifone.com.vn/web/en/services/mobiinfo.jsp

[9]. Nysveen, H., Pedersen, P.E., &Thorbjornsen, H. (2005).Intentions to Use Mobile Services: Antecendents and Cross-Service Comparisons. Academy of Marketing Science.Journal; Summer 2005; 33, 3;ABI/INFORM Global.

[10]. PC-WORLD(2012). *Mobile Malware* available at http://www.pcworld.com/mobilemalware accessed on 3rd September 2012.