Securing E-Mail Communication Using Hybrid Cryptosystem on Android-based Mobile Devices

Teddy Mantoro*[1,2], Andri Zakariya[2]
[1]Faculty of Information Technology, University of Budi Luhur (UBL)
Jalan Ciledug Raya, Petukangan Utara, Jakarta, Indonesia, Ph./Fax: +6221-5853753/5869225
[2]Advanced Informatics School, University of Technology Malaysia (UTM)
Jalan Semarak, Kuala Lumpur, Malaysia, Ph./Fax: +603-26154443
e-mail: teddy@ic.utm.my*[1], andri.zakariya@gmail.com[2]

Abstrak

Salah satu layanan internet yang paling populer adalah electronic mail (e-mail). Dengan menggunakan perangkat mobile yang terhubung dengan internet, e-mail dapat digunakan secara luas oleh berbagai kalangan untuk saling bertukar informasi dimanapun dan kapanpun baik untuk informasi yang bersifat biasa maupun yang bersifat rahasia. Sayangnya, terdapat beberapa masalah keamanan pada komunikasi menggunakan e-mail; seperti media komunikasi yang digunakan yaitu jaringan terbuka internet dan e-mail disimpan pada e-mail server yang tidak dapat dijamin keamanannya. Selain itu, e-mail tidak memiliki proteksi terhadap integritas isi pesan sehingga apabila isi pesan diubah di e-mail server atau pada saat transmisi maka tidak dapat dideteksi. E-mail juga tidak memiliki sistem otentikasi pengirim, sehingga tidak ada jaminan bahwa e-mail yang diterima oleh seseorang berasal dari pihak pemilik alamat e-mail. Untuk mengatasi permasalahan tersebut, penelitian ini mengusulkan sebuah metode pengamanan komunikasi e-mail pada perangkat mobile berbasis Android menggunakan hybrid cryptosystem dengan mengkombinasikan enkripsi simetrik, asimetrik dan fungsi hash. Hasil eksperimen menunjukkan bahwa metode yang diusulkan dapat memenuhi seluruh aspek keamanan informasi yang meliputi kerahasiaan, integritas data, otentikasi and tidak dapat dilakukan penyangkalan.

Kata kunci: *android, keamanan, e-mail, hybrid cryptosystem, integritas data*

Abstract

One of the most popular internet services is electronic mail (e-mail). By using mobile devices with internet connection, e-mail can be widely used by anyone to exchange information anywhere and anytime whether public or confidential. Unfortunately, there are some security issues with email communication; e-mail is sent in over open networks and e-mail is stored on potentially insecure mail servers. Moreover, e-mail has no integrity protection so the body can be undectected altered in transit or on the e-mail server. E-mail also has no data origin authentication, so people cannot be sure that the emails they receive are from the e-mail address owner. In order to solve this problem, this study proposes a secure method of e-mail communication on Android-based mobile devices using a hybrid cryptosystem which combines symmetric encryption, asymmetric encryption and hash function. The experimental results show that the proposed method succeeded in meeting those aspects of information security including confidentiality, data integrity, authentication, and non-repudiation.

Keywords: *android, e-mail, security, hybrid cryptosystem, data integrity*

1. Introduction

A few years ago, when people want to connect to the internet to use services such as e-mail, web browsing, chatting, and so on, they have dependency to fixed-line connections. However, as technology becomes more and more developed, people can connect to the internet without fixed-line connections anymore. It is now almost internet services can be enjoyed by using mobile devices such us notebook, smartphone and tablet PC anywhere and anytime. One of the most popular internet services is e-mail. By using mobile devices with internet connection, e-mail services can be widely used by many people to exchange information and collaborate, both for individual, enterprise and government. Knowingly, or not, the usage of e-mail to exchange information and collaborate, is not only limited to public information, but also confidential information, which has a value of confidentiality to certain parties so that it needs some security controls [1].

Received July 9, 2012; Revised September 9, 2012; Accepted September 22, 2012

In fact, e-mail is sent over open networks and e-mail is stored on potentially insecure e-mail servers. Moreover, e-mail has no integrity protection so that the contents can be undetected altered in transit or on the e-mail server, and e-mail also has no data origin authentication where people cannot be sure that email they received are from the e-mail address owner. Therefore, e-mail communication has very important security issues. In order to protect the information which is communicated via e-mail, it is require to implement all things related to the information security of the e-mail. These things include aspects of information security such as: confidentiality, data integrity, authentication, and non-repudiation [2].

Various methods have been discussed and proposed in order to provide security on e-mail communication. Unfortunately, most of them are seen to be not comprehensive enough to meet the aspects of information security. One of the researches regarding the security of e-mail communication on mobile devices has been performed using ElGamal encryption algorithm [3]. The implementation of the research was using uses Android. But, the results do not meet all aspects of information security yet, wherein that research only meets confidentiality, but does not meet data integrity, authentication, and non-repudiation aspect.

Another related research about provide security on e-mail communication using block cipher RC2 and MD5 hash function by built an add-on for Mozilla Thunderbird show that they are only meet confidentiality aspect [4]. Likewise research of secure email using block cipher XXTEA for desktop-based also have been developed as an add-in application for Microsoft Outlook 2003, but the results only meet confidentiality and data integrity aspects [5]. Moreover, some applications attempt to attack the security of e-mail communication using the adaptive chosen ciphertext [6]. However, from the results of these researches, they indicating that a mathematical attack is successful merely in securing e-mails that use the encryption mechanism only.

This paper proposes a method for securing e-mail communication on mobile devices by using the hybrid cryptosystem which is a combination of symmetric, asymmetric encryption system and hash function. Therefore, the study for securing e-mail communication that utilizes a hybrid cryptosystem on Android-based mobile devices is expected to meet all aspects of information security consists of confidentiality, data integrity, authentication, and non-repudiation. The contribution of this study is to provide a security guarantee to e-mail users from the vulnerability of using e-mail via the mobile internet.

2. Related Work

2.1. Hybrid Cryptosystem

Symmetric and asymmetric ciphers each have their own advantages and disadvantages. Symmetric ciphers are significantly faster than asymmetric ciphers, but require all parties to somehow share a secret (the key). The asymmetric algorithms allow public key infrastructures and key exchange systems, but at the cost of speed. A hybrid cryptosystem is a protocol using multiple ciphers of different types together, each to it's best advantage [7]. One common approach is to generate a random secret key for a symmetric cipher, and then encrypt this key via an asymmentric cipher using the recipient's public key. The message itself is then encrypted using the symmetric cipher and the secret key. Both the encrypted secret key and the encrypted message are then sent to the recipient.

In order to implement aspects of information security in the e-mail communication system, it will use the mechanism of the hybrid cryptosystem which combines symmetric encryption, asymmetric encryption and hash function. The cryptographic algorithms consist of AES 128 bit encryption for confidentiality, SHA 160 bit for data integrity, while aspects of authentication and non-repudiation use a combination of SHA 160 bit and RSA 1024 bit.

2.2. Advanced Encryption Standard (AES)

Advanced Encryption Standard (AES) is a specification for the encryption of electronic data. It has been adopted by the U.S. government and now it is used worldwide. The algorithm described by AES is a symmetric-key algorithm, meaning the same key is used for both encrypting and decrypting the data. In the US, AES was announced by the National Institute of Standards and Technology (NIST) as U.S. FIPS PUB 197 (FIPS 197) on November 26, 2001 after a five-year standardization process in which fifteen competing designs were presented and evaluated before it was selected as the most suitable [8].

Originally called Rijndael, the algorithm was developed by two Belgian cryptographers, Joan Daemen and Vincent Rijmen, and submitted by them to the AES selection process. The name Rijndael is a play on the names of the two inventors. AES has a fixed block size of 128 bits and a key size of 128, 192, or 256 bits, whereas Rijndael can be specified with block and key sizes in any multiple of 32 bits, with a minimum of 128 bits. The blocksize has a maximum of 256 bits, but the keysize has no theoretical maximum.

AES operates on a 4×4 column-major order matrix of bytes (versions of Rijndael with a larger block size have additional columns in the state). Most AES calculations are done in a special finite field. The AES cipher is specified as a number of repetitions of transformation rounds that convert the input plaintext into the final output of ciphertext. Each round consists of several processing steps, including one that depends on the secret key. A set of reverse rounds are applied to transform ciphertext back into the original plaintext using the same secret key. Briefly, the encryption process of AES is shown in Figure 1.

2.3. Secure Hash Algoritm – 1 (SHA-1)

Secure Hash Algorithm (SHA-1) is a further development of the MD5 hash function algorithm developed by NIST and published as FIPS 180-2 [9]. A revised version of FIPS 180-1 FIPS 180 was issued in 1995, known as SHA-1. The SHA-1 algorithm is for computing a condensed representation of a message or a data file. When a message of any length < 2^{64} bits is input, the SHA-1 produces a 160-bit output called a message digest.

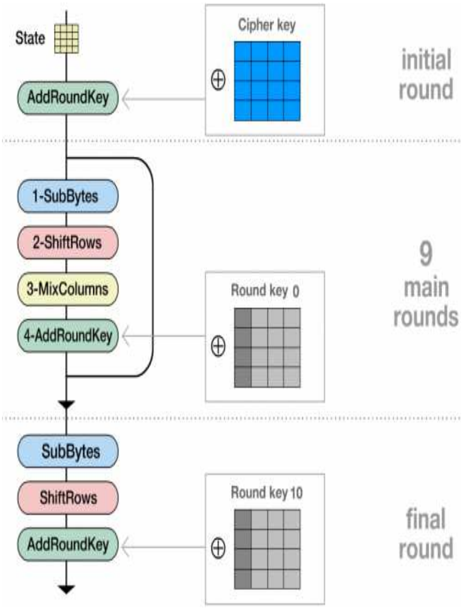


Figure 1. The AES encryption process

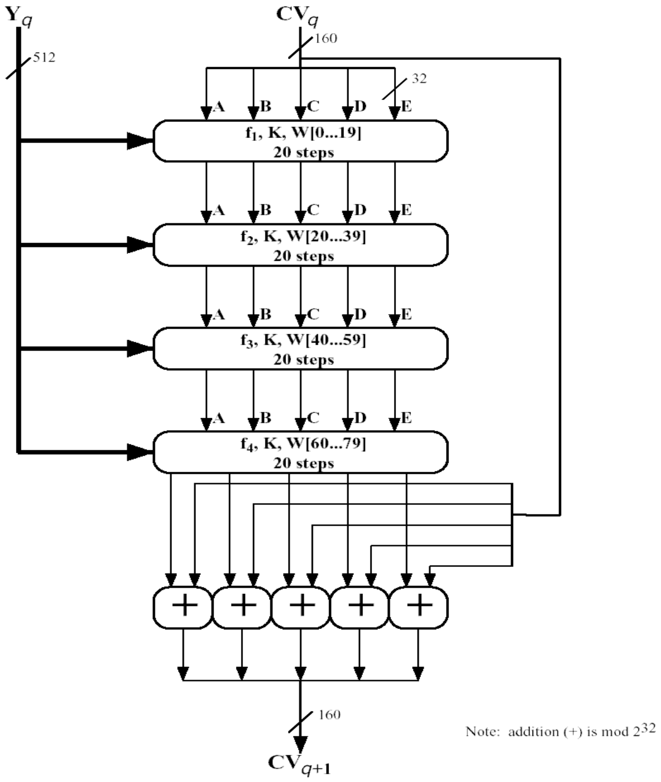

Y_q
CV_q
160
512
32
A B C D E
f_1, K, W[0...19] 20 steps
f_2, K, W[20...39] 20 steps
f_3, K, W[40...59] 20 steps
f_4, K, W[60...79] 20 steps
+
160
CV_{q+1}
Note: addition (+) is mod 2^{32}

Figure 2. SHA-1 hash function process

The message digest can then be input to the Digital Signature Algorithm (DSA) which generates or verifies the signature for the message. Signing the message digest rather than the message often improves the efficiency of the process because the message digest is usually much smaller in size than the message. The same hash algorithm must be used by the verifier of a digital signature as was used by the creator of the digital signature. In brief, the process of calculating the message digest of SHA-1 160 bit can be seen in Figure 2.

2.4. RSA

RSA is an asymmetric encryption system that is currently quite practical and universal standards made by R. Rivest, A. Shamir, and L. Adleman [2]. Its strength is emphasized by the complexity of the modulo operation with large number's reappointment. Reappointment operations used for both encryption and decryption processes are similar, the difference between the two processes lies in the numbers being used as a power. These two numbers have properties inverse to each other. The RSA encryption algorithm is described in the operation (1):

$$c = m^e \bmod n \tag{1}$$

Whereas, the decryption operation is described in the operation (2):

$$m = c^d \bmod n \tag{2}$$

c = ciphertext
m = plaintext
e = encryption key
d = decryption key
n = the multiplication of two primes

2.5. E-mail Communication Protocol

The protocol commonly used in the e-mail communication is Simple Mail Transfer Protocol (SMTP), Post Office Protocol 3 (POP3) and Internet Message Access Protocol (IMAP). The protocol used to send e-mails is SMTP, whereas the protocol used to download e-mails is POP3 and IMAP. In this paper, the protocols to be used for e-mail communication on the Android-based mobile devices are the SMTP and POP3.

2.6. Android Operating System

Android is a Linux-based operating system for mobile devices, such as mobile devices and tablet computers. It was developed by the Open Handset Alliance led by Google. Google purchased the initial developer of the software, Android Inc., in 2005. The unveiling of the Android distribution in 2007 was announced with the founding of the Open Handset Alliance, a consortium of 86 hardware, software, and telecommunication companies devoted to advancing open standards for mobile devices.

Android is a software stack for mobile devices that includes an operating system, middleware and key applications. The Android SDK provides the tools and APIs necessary to begin developing applications on the Android platform using the Java programming language. Figure 3 presented diagram of Android operating system major components [10].

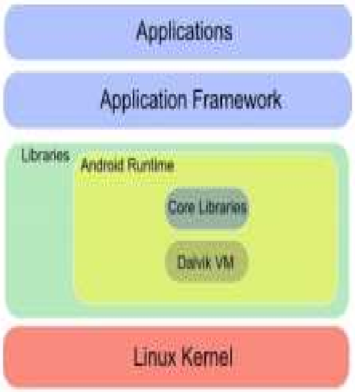


Figure 3. Android Architecture [10]

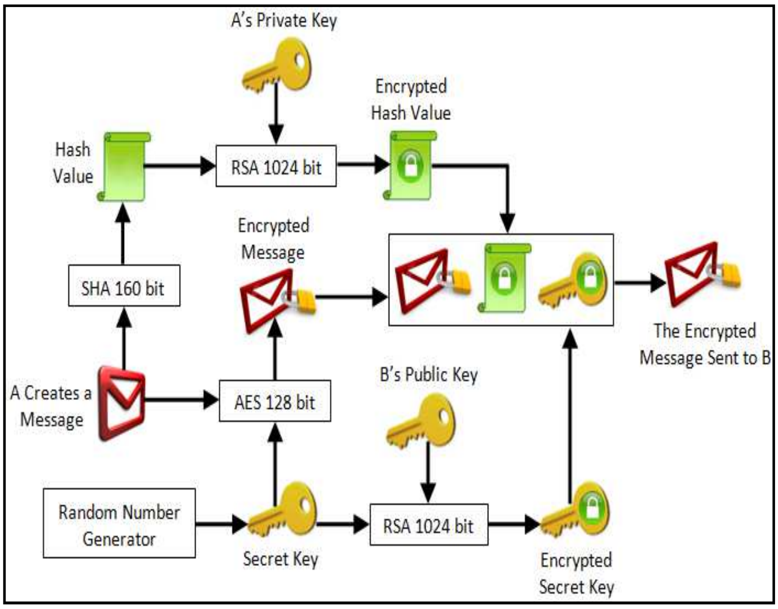


Figure 4. Secure e-mail process

3. Research Method

In this study, each client who wishes to perform secure e-mail communications should be registered on the same public-key infrastructure. On the system scenario, both public and private key are generated by each client. As they get the keys, they have to publish their public key to a public-key infrastructure server, so that other clients can download their public key from mobile device.

Both A and B are clients who wish to communicate to each other using secure e-mails. Client A will send an e-mail to client B. In brief, the mechanism of secure e-mail communication on Android-based mobile device is as the following process.

(1) Before both client A and client B can communicate to each other, they have to publish their public key on a public-key infrastructure server.
(2) The secret key is required to encrypt the e-mail using AES 128 bit. This secret key is generated randomly in a system using a random number generator.
(3) The e-mail wish to send to client B is encrypted using AES 128 bit.
(4) Due to data integrity purposes, the e-mail must be hashed using SHA 160 bit to get the message digest of the e-mail before the e-mail is encrypted.
(5) Not only the encrypted e-mail, but also the secret key and the message digest of the e-mail should be sent to client B. In order to secure the secret key and the message digest while transmitting, both the secret key and the message digest of the e-mail are encrypted using RSA 1024 bit before they are sent to client B. Technically, client A encrypts the secret key using client B's public key and client A encrypts the message digest of the e-mail using client A's private key.
(6) Then the concatenation of the encrypted e-mail, secret key and message digest of the e-mail are sent to client B through the e-mail server using SMTP.
(7) The client B downloads the packet from the e-email server using POP3.
(8) The client B sorts the packet which contains the encrypted e-mail, secret key and message digest of the e-mail.
(9) The client B decrypts the encrypted secret key with client B's private key and the encrypted message digest of the e-mail with client A's public key using RSA 1024 bit.
(10)The encrypted e-mail is also decrypted with the secret key using AES 128 bit. Due to data integrity purposes, client B calculates the message digest of the e-mail using SHA 160 bit and verifies it with the message digest they have.
(11)If client B produces the same message digest, then the e-mail is verified and there is no suspected e-mail modification while transmitting.

The secure e-mail processes before being sent to the recipient is shown in Figure 4.

4. Results and Discussion

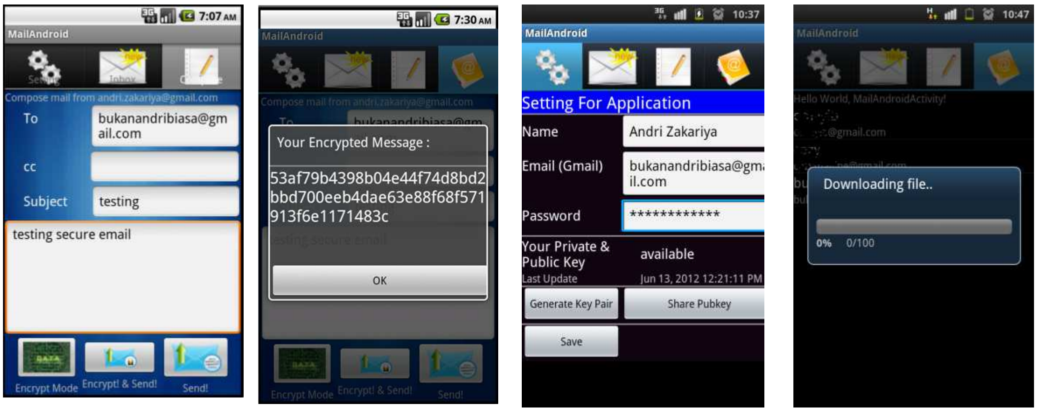
In this section, the experimental results will be discussed based on the proposed method. The experiment was conducted using Java, with Eclipse Helios and Android 2.2 SDK. The purpose of the experimental is to prove the concept that the system meets all aspects of information security including confidentiality, data integrity, authentication, and non repudiation. Figure 5 are snapshots of the application which developed on Android.

4.1. Confidentiality Testing

Confidentiality testing is done by comparing the results of the AES 128 bit encryption process, which was built on Android, with the open source AES 128 bit application software. To view the encryption results on the Android application is by performing the data packet sniffing via WiFi using Wireshark. The experiment was conducted in string data type. The result of the confidentiality tests are shown in Table 1.

4.2. Data Integrity Testing

Due to make sure and verify that the message was not altered or modified while transmission. Data integrity testing is done by comparing between the results of the SHA 160 bit hash function process which was built on Android (before the concatenation process), with the open source SHA 160 bit application software. The encryption results on the Android application are viewed by performing data packet sniffing via WiFi using Wireshark. The experiment was conducted in string data type. The results of the confidentiality testing is shown in Table 2.



Compose mail from andri.zakariya@gmail.com
To: bukanandribiasa@gmail.com
cc:
Subject: testing
testing secure email

Your Encrypted Message :
53af79b4398b04e44f74d8bd2bbd700eeb4dae63e88f68f571913f6e1171483c

Setting For Application
Name: Andri Zakariya
Email (Gmail): bukanandribiasa@gmail.com
Password: ************
Your Private & Public Key: available
Last Update: Jun 13, 2012 12:21:11 PM

Hello World, MailAndroidActivity!
Downloading file..
0% 0/100

(a) (b) (c) (d)

Figure 5. Compose message (a), After sending the message (b), Setting screen (c), and Downloading public key screen (d)

Table 1. Confidentiality testing result

Plaintext	Key	Ciphertext		Result
		Android	Software	
My credit card number is 1234 5678 1234 5678	939034839d5e51b337ee8a6b9e58a887	x00EcCrMtcM94n1sP33IKOTef3czUpLGKvhj+xjPrbx4bDSaTkXlLkf7gc2I1pPquUBziH6Dv7oGp7dUpr5Dnw==	x00EcCrMtcM94n1sP33IKOTef3czUpLGKvhj+xjPrbx4bDSaTkXlLkf7gc2I1pPquUBziH6Dv7oGp7dUpr5Dnw==	Match
The secret operation Alpa-1 will be done by tommorow	c980ef30c9b85b008c3dd633a427ef18	wsESNaI9QOomd11H70aqB2722Lx0mMsP+JI/yn/0PxwNMFZROY12ruePV+eochUIrBgpTNQ22z3eBUFkccmsUk/AEx1G7dpmt9BT+a1hT1c=	wsESNaI9QOomd11H70aqB2722Lx0mMsP+JI/yn/0PxwNMFZROY12ruePV+eochUIrBgpTNQ22z3eBUFkccmsUk/AEx1G7dpmt9BT+a1hT1c=	Match
The next target is Ritz Carlton	b604bb4729b319c7007ae3f18fcf2951	16VJh3CZzh0Een8obuJEWe88xj68R4H5inwa1an6M9GiiYDUAPyWiSGpNX9WikES	16VJh3CZzh0Een8obuJEWe88xj68R4H5inwa1an6M9GiiYDUAPyWiSGpNX9WikES	Match
Bunderan HI akan kita jadikan lautan api	c2763f08d5ece2523bad0db867c016cb	6Hxnumg6aSjJysGVMnyxGEBQtKD8fK22P2w/3ivZD6/hCEljxC5Ozzjor1krQcU4J+uozRU1ZFG6FqSzZL+ttQ==	6Hxnumg6aSjJysGVMnyxGEBQtKD8fK22P2w/3ivZD6/hCEljxC5Ozzjor1krQcU4J+uozRU1ZFG6FqSzZL+ttQ==	Match
Temui saya di depan EX Plaza nanti malam	f0db1c8c14b8377266be746cd8fdf07a	FIwnIZR7arg7FqduHqpUSnrvGzNZK7ajBNHJWWkcrFCpBb9JFBnU2EbQPvHZBxi86kT6uJQPSHl/3QJ5PAvgHg==	FIwnIZR7arg7FqduHqpUSnrvGzNZK7ajBNHJWWkcrFCpBb9JFBnU2EbQPvHZBxi86kT6uJQPSHl/3QJ5PAvgHg==	Match

Table 2. Data integrity testing result

Message	Message digest		Result
	Android	Software	
My credit card number is 1234 5678 1234 5678	GDgO0IJBqR9cDj6HCLJTI1nydiE=	GDgO0IJBqR9cDj6HCLJTI1nydiE=	Match
The secret operation Alpa-1 will be done by tomorrow	fof+Jexv+AOglF15E96P3gcf5NI=	fof+Jexv+AOglF15E96P3gcf5NI=	Match
The next target is Ritz Carlton	9DHZ6+7lo3OdlihuJGd73SBteXY=	9DHZ6+7lo3OdlihuJGd73SBteXY=	Match
Bunderan HI akan kita jadikan lautan api	J5Klw3okRZqfxXDT0HXQh+LGBoo=	J5Klw3okRZqfxXDT0HXQh+LGBoo=	Match
Temui saya di depan EX Plaza nanti malam	X+akZMOkfg4fqukcpBoDJAda3/M=	X+akZMOkfg4fqukcpBoDJAda3/M=	Match

4.3. Authentication and Non-Repudiation Testing

Combination of SHA 160 bit and RSA 1024 bit are an improvement to security by providing authentication and non-repudiation on e-mail communications. The message digest of message is be encrypted by RSA using the sender’s private key. Then the receiver decrypts it using the sender’s public key and calculates the message digest of the message. The receiver will verify the message digest by comparing it with the message digest of the message which the receiver decrypted using AES 128 bit before. The result is the receiver gets the verified message digest.

Anyone who has the sender’s public key could decrypt the message digest of the message and get the verified message digest, then the sender is authenticated and cannot argue if the sender has not confessed to sending the message.

5. Conclusion

In this paper a method for securing e-mail communication on Android-based mobile device is introduced. The proposed method uses hybrid cryptosystem which combines symmetric encryption, asymmetric encryption and hash function consisting of AES 128 bit, RSA 1024 bit and SHA 160 bit. The advantage of this technique meets the aspects of information security including confidentiality, data integrity, authentication, and non repudiation between two communication parties. The experimental results show that the system succeeded in meeting those aspects of information security.

Several recommendations are suggested to improve this system so that it can provide more features. The recent system is able to send string data type only what is typed in the email interface. For future work, this system could provide encryption for email attachments and interoperability with other platforms, such as desktops or other mobile device platforms.

References

[1] Li B, Im EG. Smartphone, Promising Battlefield for Hackers. *Journal of Security Engineering*. 2011; 8(1): 89-110.

[2] Menezes AJ, Oorschot PC, Vanstone SA. Handbook of Applied Cryptography. Florida: CRC Press Inc. 1996.

[3] Adinagara YT, Winarno I, Fathoni K. *Enkripsi E-mail Dengan Menggunakan Metode ElGamal pada Perangkat Mobile*. Institut Teknologi Sepuluh Nopember. Report number: 1228. 2011.

[4] Fernando RG. Pembangunan Add-On Pada Mozilla Thunderbird Untuk Enkripsi Surat Elektronik Dengan Corrected Block Tiny Encryption Algorithm. Bachelor Thesis. Bandung: Institut Teknologi Bandung; 2009.

[5] Yusoff MA. Secure Email. Master Thesis. Johor: Universiti Teknologi Malaysia; 2008.

[6] Parashar R, Parihar PS, Kurdia V. A Attack on E-Mail Encryption Protocols by Chosen Ciphertext Method. *International Journal of Internet Computing*. 2011; 1(1): 65-68.

[7] Ramaraj E, Karthikeyan S, Hemalatha M. A Design of Security Protocol using Hybrid Encryption Technique (AES- Rijndael and RSA). *International Journal of The Computer, the Internet and Management*. 2009; 17(1): 78-86.

[8] Federal Information Processing Standards Publications. 197. *Advanced Encryption Standard*. Washington DC: FIPS PUBS; 2001.

[9] Federal Information Processing Standards Publications. 180-2. *Secure Hash Standard*. Washington DC: FIPS PUBS; 2002.

[10] Gandhewar N, Sheikh R. Google Android: An Emerging Software Platform For Mobile devices. *International Journal on Computer Science and Engineering*. 2010; 1(1): 12-17.

[11] Bharati JM, Hemalatha S, Aiswarya V. Advancement in Mobile Communication using Android. *International Journal of Computer Applications*. 2010; 1(7): 95-98.

[12] Pandove K, Jindal A, Kumar R. Email Security*. International Journal of Computer Applications.* 2010; 5(1): 23-26.

[13] Banerjee U, Vashishtha A, Saxena M. Evaluation of the Capabilities of WireShark as a Tool for Intrusion Detection. *International Journal of Computer Applications.* 2010; 6(7):1-5.

[14] Bang H, Noh B. Design Approaches of Android for Students. *International Journal of Computer Science and Network Security*. 2010; 10(12): 225-230.

[15] Gill S, et.al. Email Security Protocol. *International Journal of Computer Trends and Technology.* 2011; 1(1): 1-5.

[16] Seth SM, Mishra R. Comparative Analysis of Encryption Algorithms for Data Communication. *International Journal of Computer Science and Technology.* 2011; 2(2): 292-294.

[17] Bonner E, O'Raw J, Curran K. Implementing the Payment Card Industry (PCI) Data Security Standard (DSS). *TELKOMNIKA: Indonesian Journal of Electrical Engineering*. 2011; 9(2): 365-376.

[18] Mullally A, McKelvey N, Curran K. Performance Comparison of Enterprise Applications on Mobile Operating Systems. *TELKOMNIKA: Indonesian Journal of Electrical Engineering*. 2011; 9(3): 503-514.

[19] Torkaman MRN, Kazazi NS, Rouddini A. Innovative Approach to Improve Hybrid Cryptography by Using DNA Steganography. *International Journal on New Computer Architectures and Their Applications*. 2012; 2(1): 225-236.

[20] Yoon HJ. A Study on the Performance of Android Platform*. International Journal on Computer Science and Engineering*. 2012; 4(4): 532-537.

[21] Rasmi PS, Paul V. *A Hybrid Crypto System based on a new CircleSymmetric key Algorithm and RSA with CRT Asymmetric key Algorithm for E-commerce Applications*. International Conference on VLSI, Communication & Instrumentation*.* Kerala. 2011; 9: 14-18.

[22] Speckman B. The Android Mobile Platform. Master Thesis. Michigan: Eastern Michigan University; 2008.

[23] Schneier B. Applied Cryptography. Second Edition. New York: Wiley & Sons, Inc. 1995.