International Journal of Control and Automation
Vol.10, No.3 (2017), pp.345-356
http://dx.doi.org/10.14257/ijca.2017.10.3.28

The Combining Method of Fingerprint and QR Code as Mutual Authentication for Mobile Payment

Ariana Tulus Purnomo[1, 2], Chang-Soo Kim[2], Yudi Satria Gondokaryono[1] and Ilkyeun Ra[3]

[1]Departement of Electrical Engineering,
Institut Teknologi Bandung, Bandung, 40132, Indonesia
[2]Departement of IT Convergence and Application Engineering,
Pukyong National University, Busan, 608-737, South Korea
[3]Departement of Computer Science and Engineering,
University of Colorado Denver, Denver, CO 80203, United States of America
tulusariana@gmail.com[1, 2], cskim@pknu.ac.kr[2], ygondokaryono@stei.itb.ac.id[1], ilkyeun.ra@ucdenver.edu[3]

Abstract

The existing payment system using a debit/ credit card is not safe. The cards are easy to lose, and sometimes people forget to bring card while doing a transaction. Nowadays, payment system has shifted to the digital world. Most people never leave their mobile phones. This situation shows that there is significant opportunity to use mobile devices as mobile commerce applications, especially for paying through the use of credit cards. Unfortunately, the mobile device is still not secure and enable the variety of security risks. In this paper, we propose the protecting method of Fingerprint and QR Code as mutual authentication for the mobile payment system. QR Code and Fingerprint are combined to make mobile payment system more secure than ever. Because each of our fingers has their unique pattern, we can bring an additional layer of protection for authentication by making a sequence of which finger should be read first, second, and so on. The adoption of QR Code ensures the legibility, integrity, and confidentiality of the transmitted information, and lays a good foundation for the overall security of the system.

Keywords*: fingerprint, QR Code, Mobile payment security, RSA.*

1. Introduction

By using a smartphone, some activities become faster and scalable. Currently, mobile commerce provides a variety of service such as mobile ticketing, mobile banking, mobile location-based service, mobile auctions, mobile purchasing and so on. This situation shows that there is significant opportunity to use mobile devices as mobile commerce applications, especially for paying through the use of credit cards.

Many people develop innovations to improve the shopping experience. Rise capability of mobile devices brings through payment system into mobile devices. If our credit/debit card is lost, it has a significant risk for our money to be stolen. By using mobile payment system, people do not need to carry credit/debit card for the transaction. Besides that, if someone forgets to bring credit/debit card, it is not a problem anymore.

Proposed mobile payment system acts as a solution for the mobile payment service provider that provides the seller and the buyer to confirm payment to the bank account. However, using a mobile payment system allow risk to the cardholder, so securing account data is also important for reducing risks that may occur to the cardholder.

[*]Corresponding Author: Chang-Soo Kim(cskim@pknu.ac.kr)

2. Mobile Payment Security

The internet is used in various fields. By using the internet, it will simplify payment solution. Unfortunately, the internet is not safe for private commercial transactions. During related transaction using mobile payment, all data should be secured.

A mobile device is created to make it easier to be used. Security improvements on a mobile device cause a trade-off in ease of use. However, security is the most important part of the mobile payment system to gain client confidence in the market. Payment system involves several environments such as sellers, device, operating system developers, application design, network operator and some protocol that connected to a different entity. More secure mobile payment system it means many people will trust it in the market.

Unfortunately, a mobile device is still not secure and enable a variety of security risks. There are several techniques to secure mobile payment system such as using passcodes, pin number of the card, identification of the serial number, etc. However, this security system is still not effective because it is easy to be replicated. Many possible attacks targeting mobile payment transactions will continue to increase. There will be risks associated with them. Secure mobile payments that are made today may not be secure in the future because there is much- uncovered reason. Because of that reason, in this paper, we propose one more security level mechanism that could be used. Logical device access protecting methods such as biometrics, complex passwords, multi-factor authentication can be used to enhance the security of mobile payment application. The fingerprint, a fingerprint is one that cannot be replicated [4].

During use of mobile payment, three major risks need to be protected: before data enters the device, processes in the device and leaves the device. Mobile payment service providers have an obligation to protect cardholder data such as protect any payment card information, whether it is printed, processed, transmitted or stored. Cardholder data should be encrypted before entering the mobile device, processing in the mobile device and transmitting to the network. It is a challenge in the securing environment for mobile payment service providers.

To protect a user’s account data from an intruder, all parties that involve during the transaction should cooperate keeping the environment. The client has significant responsibilities to keep their account data, such as keep save the password when entering it into a device. The client should make sure there is no other person around him. Mobile payment application should not be able to remember passwords. Besides that, the application developer should ensure that the mobile payment application cannot be accessed by external devices such as card readers and so on. During data transmission in the network, the connection should be secure, and data must be encrypted. Login period on a mobile device should be limited to avoid risks when the user forgets to log out or the user loses a mobile device. This configuration of authentication method prevents the unwanted user to force the user re-authenticating to the device after a specified amount of time.

3. Design System

This mobile payment uses biometric and QR Code reader as the authentication mechanism. This application would be used in devices that support fingerprint scanner and camera. Here is the general design of Secure Mobile Payment System.

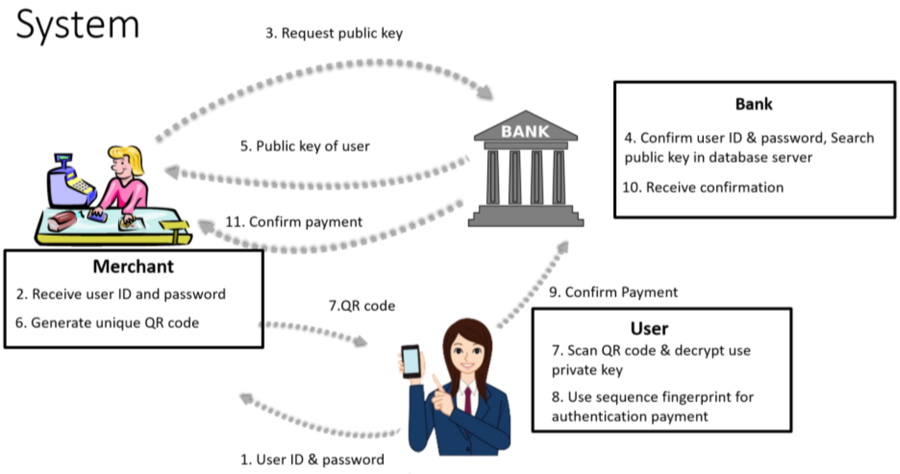

System
3. Request public key
Bank
4. Confirm user ID & password, Search public key in database server
10. Receive confirmation
BANK
5. Public key of user
11. Confirm payment
Merchant
2. Receive user ID and password
6. Generate unique QR code
7.QR code
9. Confirm Payment
User
7. Scan QR code & decrypt use private key
8. Use sequence fingerprint for authentication payment
1. User ID & password

Figure 1. Design of Mobile Payment System

Before using the system, each user must have an account on the server. Any person who wants to get access must be authenticated first. Here, we assume that all users already have an account on the server. It includes the user personally identifiable information such as his fingerprint, user ID, and password.

In this system, we have three parties involved. They are the User, the Merchant, and the Bank/Server. The flowchart of a user mobile payment application system can be seen in Figure 2.

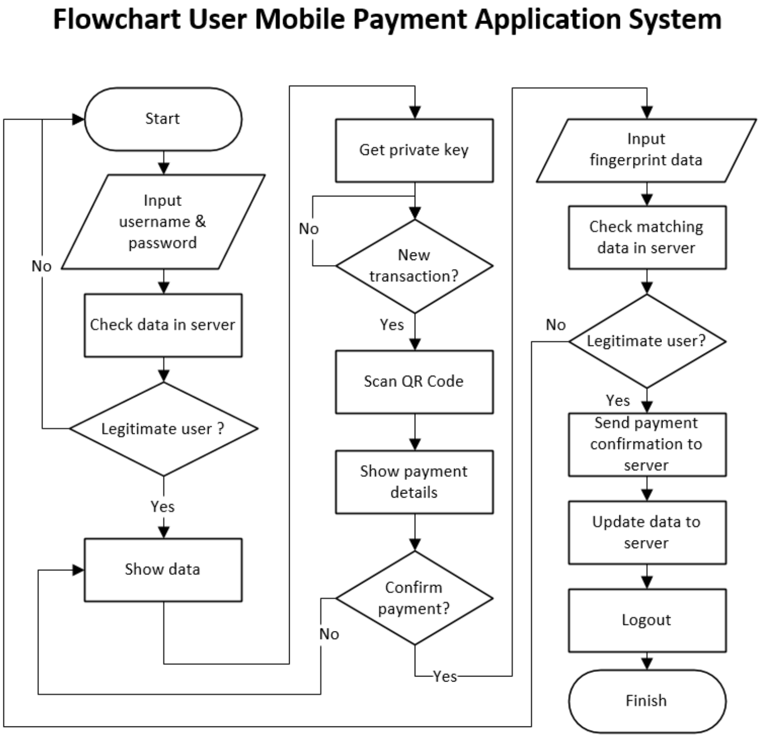


Figure 2. Flowchart User Mobile Payment Application System

The flowchart of merchant application system can be seen in Figure 3.

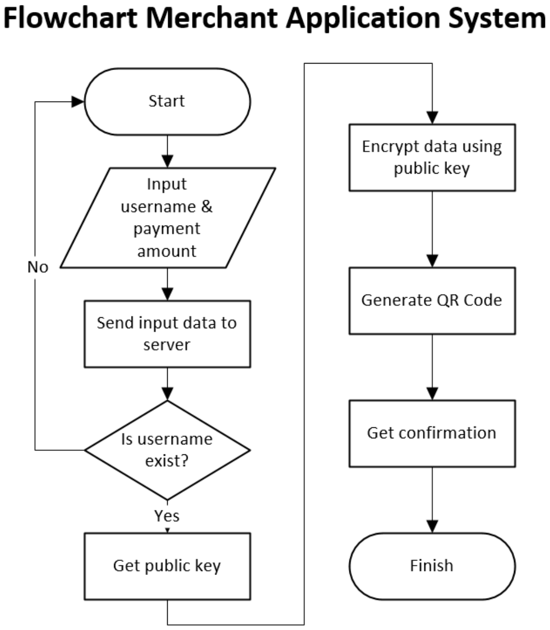

Flowchart Merchant Application System
Start
Input username & payment amount
Send input data to server
Is username exist?
No
Yes
Get public key
Encrypt data using public key
Generate QR Code
Get confirmation
Finish

Figure 3. Flowchart Merchant Application System

The flowchart of server system can be seen in Figure 4.

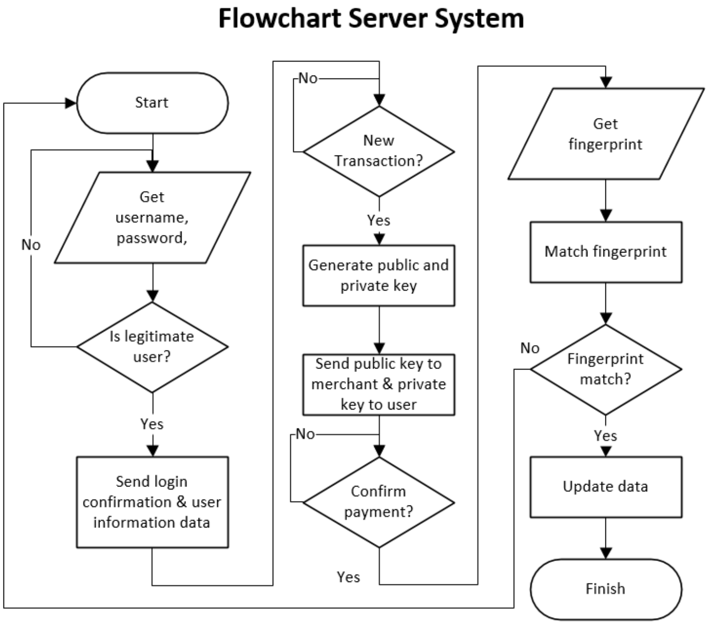


Figure 4. Flowchart Server System

In this mobile payment transaction, first, the user logins using the user ID in the merchant’s device. The merchant will send a request to the third party to generate the user’s public key. The third party generates user’s public key and private key based on the identity of the user. After the public key and the private key are generated, then the public key is sent to the merchant and the private key is submitted to the user.

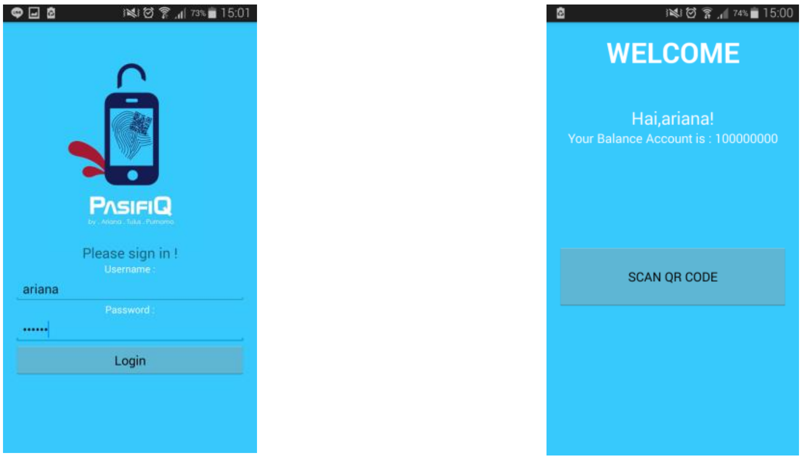
After merchant receives the public key, merchant's device encrypts transaction information using RSA algorithm. This method will protect the user from wrong withdrawal party and also safeguard the merchant from fraud parties which have a likelihood to receive payment confirmation. After the information is encrypted, then the information is embedded into QR Code and ready to be scanned by users. When the merchant generates QR Code, there is a limit period to scan the QR Code. If the user does not confirm the payment during the period, then the QR Code will be expired. In this method, only the user that has the private key may confirm the payments and after confirming the payment, the QR Code cannot be used anymore. In this payment system, digital signature no longer needed because the authentication has been done by the third party that ensures legitimation of users.

To receive the private key, the user should log in to the application on his mobile phone. These two ways of authentication method will ensure that the user is a legitimate party. Then mobile payment needs to scan QR Code that is generated by the merchant. After mobile application scans the QR Code and gets the amount that should be paid, a user should confirm the payment using his account by scanning his fingerprint in the mobile device. This mobile payment application is using five combinations of the fingerprint. The fingerprint sequence method scans fingerprint five times. In this approach, user should remember which finger should be read first, second, and so on. A user should remember the sequence queue of his fingerprint as his password to confirm the payment. Then, fingerprint template of a user will be sent to the server. The server will match the fingerprint template to the fingerprint on the database. This matching process is done on the server. If fingerprint matching process and combination fingerprint match, the server will send the confirmation to the merchant and the user that the payment is accepted.

4. Implementation

We have implemented some module using C# programming language, Visual Basic – Visual Studio integrated development environment and SQL Server Database System. We also employ an open source library (ZBar Code) to encode and decode QR Code and Source AFIS for fingerprint matching. Here is the following explanation of the implementation.

4.1. Login



Please sign in !
Username : ariana
Password : ••••••

Hai,ariana!
Your Balance Account is : 100000000

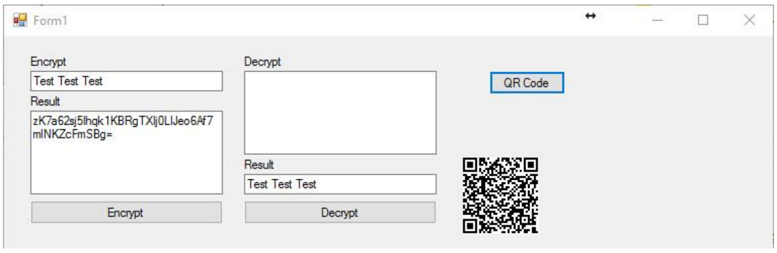
(a) (b)

Figure 5. Mobile Payment Login

4.2. QR Code

Quick Response Code (QR Code) is an array of bits that can be used storing information. QR Code that generated by the merchant act as the virtual account number for payment. Moreover, this protection method adopts QR Code to hold and transmit both authentication and authorization information. The storage capacity of the QR Code makes it capable of encapsulating the required information. The adoption of QR Code ensures the legibility, integrity, and confidentiality of the transmitted information, and lays a good foundation for the overall security of the system. QR Code generator and QR Code reader are needed to implement in the mobile payment system.

QR Code Generator. Here we implement QR Code Generator in Visual Studio using Visual Basic.NET.



Form1

Encrypt: Test Test Test

Result: zK7a62sj5lhqk1KBRgTXlj0LIJeo6Af7mINKZcFmSBg=

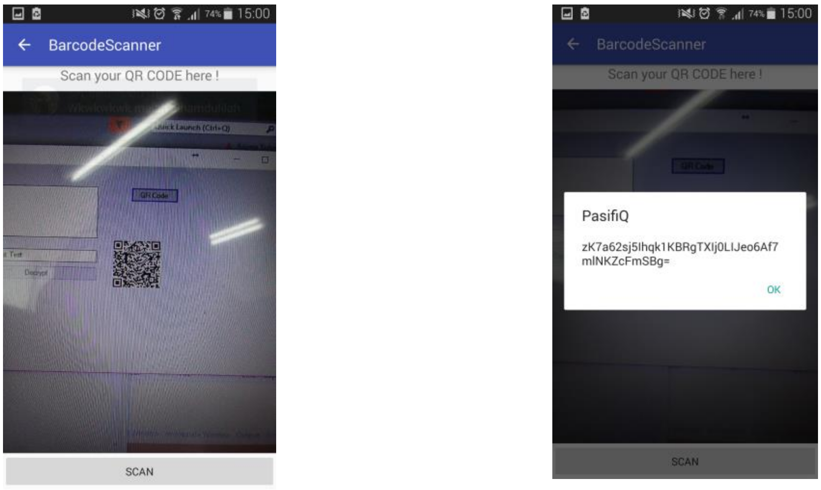
Decrypt:

Result: Test Test Test

Figure 6. Encrypt Data and Generate in QR Code

Data encoded in QR Code cannot be read directly by the human eye. QR Code scanner are needed to read data inside it. However, this is still not enough to hide the information. That is why asymmetric encryption is added to enhance the confidentiality of data.

QR Code Reader. In this mobile payment implementation, we use ZBar barcode scanner. ZBar is one of the most prominent free barcode scanner available. ZBar is an open source library software suite for scanning and reading barcodes and QR Code from various sources. This library is called in the Android Studio by Java programming language. With the ZBar library, scanning Barcodes / QR codes is quite simple. ZBar can identify multiple barcode /QR code types and able to give the cords of their locations.



BarcodeScanner

Scan your QR CODE here !

PasifiQ

zK7a62sj5lhqk1KBRgTXlj0LIJeo6Af7mINKZcFmSBg=

(a)

(b)

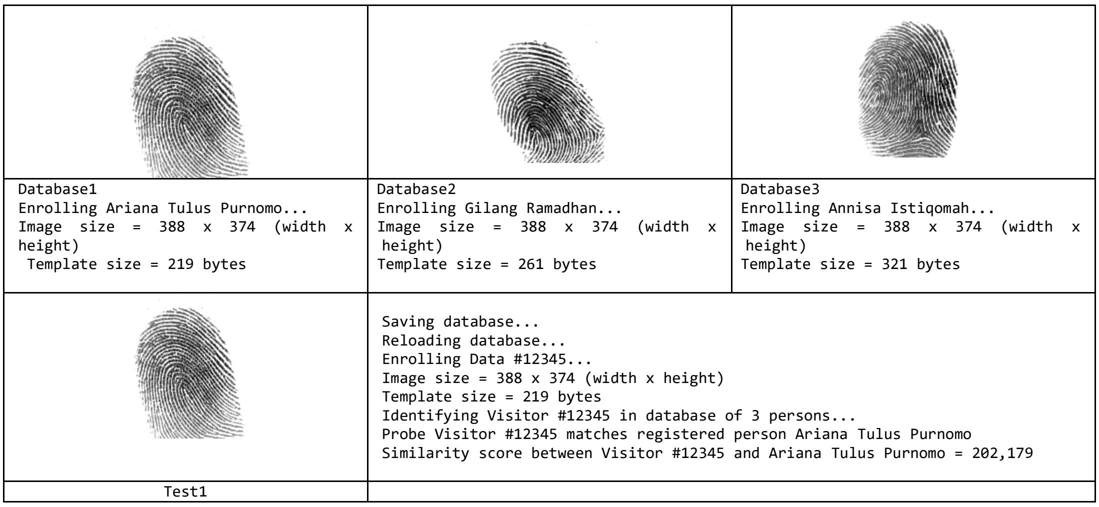
Figure 7. Implementation of QR Code Reader

4.3. Fingerprint

Nowadays smartphone comes with fingerprint scanners. The fingerprint is a powerful mechanism for biometric authentication. An authentication system uses as a means to identify and authenticate users in the mobile payment [4]. This paper proposes to use Fingerprint as the second authentication mechanism to improve the existing mobile payment system. Any authentication processing information in the mobile device is not intended. A mobile payment system will ensure end-to-end encryption of all communication between all parties. Between fingerprint template and fingerprint in the database will be not compared in a raw state. Fingerprint data that is stored on the server will also not be stored as raw data. Via biometric reader, the user sends his fingerprint to the authenticator, a software module in the server. Then, the authenticator queries the fingerprint in the registration database. If there is a match, the authenticator sends a response and forwards it to the user.

A software module in the server builds using open source library SourceAFIS. SourceAFIS is fingerprint recognition toolkit. The input data that is used for testing is fingerprint NIST, Simple_FingerPrint_Matching, FVC2002, DB1_B. This database contains an 8-bit grayscale image of the randomly selected fingerprint. This input has image format ".tif"

Here, three fingerprints are used for database testing. Then, one fingerprint is tested to match to database data. Here is the result.



```
Database1
Enrolling Ariana Tulus Purnomo...
Image  size  =  388  x  374  (width  x
height)
 Template size = 219 bytes
```

```
Database2
Enrolling Gilang Ramadhan...
Image  size  =  388  x  374  (width  x
 height)
Template size = 261 bytes
```

```
Database3
Enrolling Annisa Istiqomah...
Image  size  =  388  x  374  (width  x
 height)
Template size = 321 bytes
```

```
Saving database...
Reloading database...
Enrolling Data #12345...
Image size = 388 x 374 (width x height)
Template size = 219 bytes
Identifying Visitor #12345 in database of 3 persons...
Probe Visitor #12345 matches registered person Ariana Tulus Purnomo
Similarity score between Visitor #12345 and Ariana Tulus Purnomo = 202,179
```

Figure 8. Fingerprint Matching

Based on research before, we use a percentage of matching 97%. If there is a 97% match, it is assumed that this is the fingerprint of the customer [4]. From the fingerprint test result, it can be seen that the fingerprint input can be detected as one of the client members. This client is already registered in the database.

After the first fingerprint of user matched to the database, the server checks second, third, fourth and fifth fingerprint of the user. The first fingerprint is used for identification, second until fifth is used as the password.

4.4. Rivest Shamir Adleman (RSA)

Asymmetric Keys is two related keys, a public key and a private key that are used to perform complementary operations, such as encryption and decryption. RSA is one of the asymmetric encryption that provides the strongest known method of security.

In this implementation, RSA is used as the encryption and decryption algorithm. Where Public Key Infrastructure (PKI) is used as the key distribution mechanism. The third party is involved in generating the key pair of the public and private key. RSA private key consists of the following component.

- Modulus (the modulus n)
- Exponent (the public exponent e)
- Prime p (the prime factor p of n)
- Prime q (the prime factor q of n)
- Prime exponent p (d mod (p-1))
- Prime exponent q (d mod (q-1))
- Inverse q (the Chinese remainder theorem coefficient q - 1 mod p)

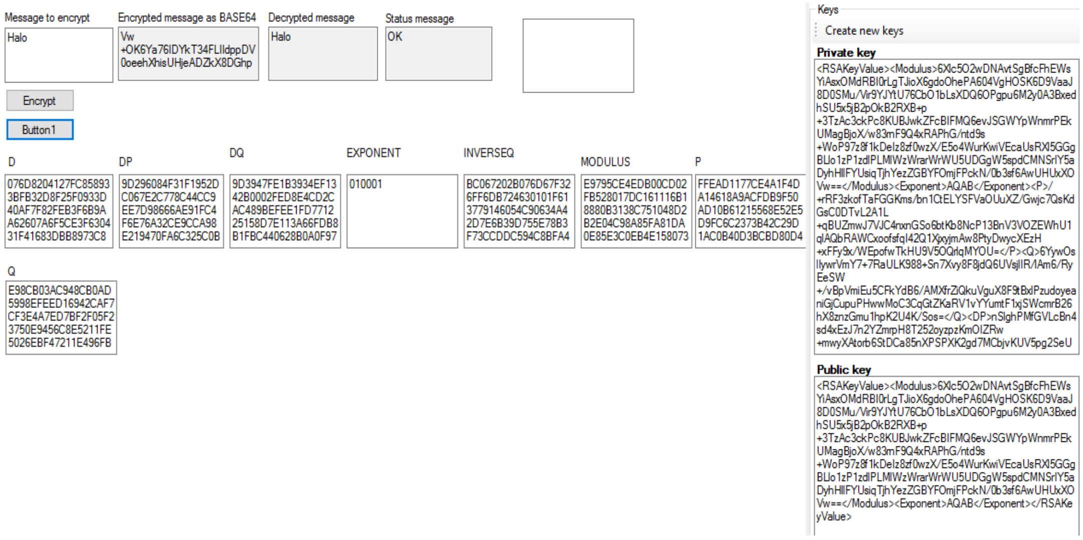
RSA public key consists of the following component.

- Modulus (the modulus n)
- Exponent (the public exponent e)

The server will generate an element of both public key and private key. Then, the component of public key will be sent to the merchant, and the component of private key will be forwarded to user's mobile payment application. User's mobile device will generate the private key from the private key component that has been received. Merchant also does the same. By sending the component of the key, the public and private key are not visible directly to the intruder. Also, these key elements are also transmitted in encrypted form.

RSA is a cipher in which the plaintext and ciphertext are integers between 0 and n-1 for some n. RSA makes use of an expression with exponentials. Plaintext is encrypted in blocks with each block having a binary value less than some number n. Public key is PU = {e, n} and a private key is PR = {d, n}. The encryption form is C = M^e mod n and decryption form is M = C^d mod n = $(M^e)^d$ mod n, where M is plaintext and C is ciphertext. Receiver and sender must know the value of n. The sender knows the value of e, and only the receiver knows the value of d.

Here is testing of public and private key generator in the server. We implement it in Visual Studio using Visual Basic.NET program language.



Message to encrypt
Halo

Encrypted message as BASE64
```
Vw
+OK6Ya76IDYkT34FLIldppDV
0oeehXhisUHjeADZkX8DGhp
```

Decrypted message
Halo

Status message
OK

D
```
076D8204127FC85893
3BFB32D8F25F0933D
40AF7F82FEB3F6B9A
A62607A6F5CE3F6304
31F41683DBB8973C8
```

DP
```
9D296084F31F1952D
C067E2C778C44CC9
EE7D98666AE91FC4
F6E76A32CE9CCA98
E219470FA6C325C0B
```

DQ
```
9D3947FE1B3934EF13
42B0002FED8E4CD2C
AC489BEFEE1FD7712
25158D7E113A66FDB8
B1FBC440628B0A0F97
```

EXPONENT
```
010001
```

INVERSEQ
```
BC067202B076D67F32
6FF6DB724630101F61
3779146054C90634A4
2D7E6B39D755E78B3
F73CCDDC594C8BFA4
```

MODULUS
```
E9795CE4EDB00CD02
FB528017DC161116B1
8880B3138C751048D2
B2E04C98A85FA81DA
0E85E3C0EB4E158073
```

P
```
FFEAD1177CE4A1F4D
A14618A9ACFDB9F50
AD10B61215568E52E5
D9FC6C2373B42C29D
1AC0B40D3BCBD80D4
```

Q
```
E98CB03AC948CB0AD
5998EFEED16942CAF7
CF3E4A7ED7BF2F05F2
3750E9456C8E5211FE
5026EBF47211E496FB
```

Private key

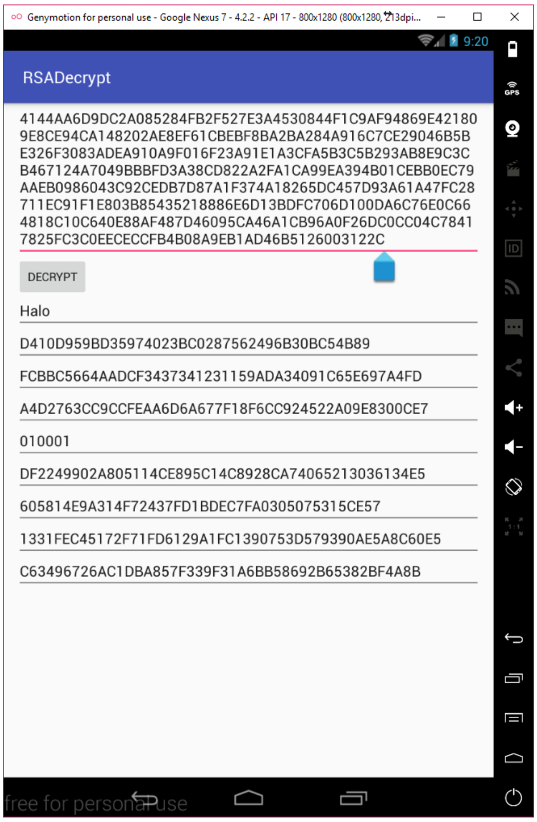
```
<RSAKeyValue><Modulus>6Xlc5O2wDNAvtSgBfcFhEWs
YiAsxOMdRBI0rLgTJioX6gdoOhePA604VgHOSK6D9VaaJ
8D0SMu/Vir9YJYtU76CbO1bLsXDQ6OPgpu6M2y0A3Bxed
hSU5x5jB2pOkB2RXB+p
+3TzAc3ckPc8KUBJwkZFcBIFMQ6evJSGWYpWnmrPEk
UMagBjoX/w83mF9Q4xRAPhG/ntd9s
+WoP97z8f1kDeIz8zf0wzX/E5o4WurKwiVEcaUsRXI5GGg
BLlo1zP1zdIPLMIWzWrarWrWU5UDGgW5spdCMNSrIY5a
DyhHlIFYUsiqTjhYezZGBYFOmjFPckN/0b3sf6AwUHUxXO
Vw==</Modulus><Exponent>AQAB</Exponent><P>/
+rRF3zkofTaFGGKms/bn1CtELYSFVaOUuXZ/Gwjc7QsKd
GsC0DTvL2A1L
+qBUZmwJ7VJC4nxnGSo6btKb8NcP13BnV3VOZEWhU1
qIAQbRAWCxoofsfqI42Q1XjxyjmAw8PtyDwycXEzH
+xFFy9x/WEpofwTkHU9V5OQrlqMYOU=</P><Q>6YywOs
IlywrVmY7+7RaULK988+Sn7Xvy8F8jdQ6UVsjlIR/IAm6/Ry
EeSW
+/vBpVmiEu5CFkYdB6/AMXfrZiQkuVguX8F9tBxlPzudoyea
niGjCupuPHwwMoC3CqGtZKaRV1vYYumtF1xjSWcmrB26
hX8znzGmu1hpK2U4K/Sos=</Q><DP>nSlghPMfGVLcBn4
sd4xEzJ7n2YZmrpH8T252oyzpzKmOIZRw
+mwyXAtorb6StDCa85nXPSPXK2gd7MCbjvKUV5pg2SeU
```

Public key

```
<RSAKeyValue><Modulus>6Xlc5O2wDNAvtSgBfcFhEWs
YiAsxOMdRBI0rLgTJioX6gdoOhePA604VgHOSK6D9VaaJ
8D0SMu/Vir9YJYtU76CbO1bLsXDQ6OPgpu6M2y0A3Bxed
hSU5x5jB2pOkB2RXB+p
+3TzAc3ckPc8KUBJwkZFcBIFMQ6evJSGWYpWnmrPEk
UMagBjoX/w83mF9Q4xRAPhG/ntd9s
+WoP97z8f1kDeIz8zf0wzX/E5o4WurKwiVEcaUsRXI5GGg
BLlo1zP1zdIPLMIWzWrarWrWU5UDGgW5spdCMNSrIY5a
DyhHlIFYUsiqTjhYezZGBYFOmjFPckN/0b3sf6AwUHUxXO
Vw==</Modulus><Exponent>AQAB</Exponent></RSAKe
yValue>
```

Figure 9. Key Pair Generator in Server

Mobile payment application also should be able to generate the private key to decrypt message inside QR Code. It is implemented in Java environments. Java security provides a key generator that can generate a key pair for public key and private key using the RSA algorithm. Key pair generator has a strength factor and a random number generator parameter. This random generator is pseudo-random number generator provided in the Java. Here is the testing of private key generator and RSA decryption in Android. We implement it in Android Studio using Java.


Genymotion for personal use - Google Nexus 7 - 4.2.2 - API 17 - 800x1280 (800x1280, 213dpi...
9:20
RSADecrypt
DECRYPT
free for personal use

```
4144AA6D9DC2A085284FB2F527E3A4530844F1C9AF94869E42180
9E8CE94CA148202AE8EF61CBEBF8BA2BA284A916C7CE29046B5B
E326F3083ADEA910A9F016F23A91E1A3CFA5B3C5B293AB8E9C3C
B467124A7049BBBFD3A38CD822A2FA1CA99EA394B01CEBB0EC79
AAEB0986043C92CEDB7D87A1F374A18265DC457D93A61A47FC28
711EC91F1E803B85435218886E6D13BDFC706D100DA6C76E0C66
4818C10C640E88AF487D46095CA46A1CB96A0F26DC0CC04C7841
7825FC3C0EECECCFB4B08A9EB1AD46B5126003122C
```

```
Halo
D410D959BD35974023BC0287562496B30BC54B89
FCBBC5664AADCF3437341231159ADA34091C65E697A4FD
A4D2763CC9CCFEAA6D6A677F18F6CC924522A09E8300CE7
010001
DF2249902A805114CE895C14C8928CA74065213036134E5
605814E9A314F72437FD1BDEC7FA0305075315CE57
1331FEC45172F71FD6129A1FC1390753D579390AE5A8C60E5
C63496726AC1DBA857F339F31A6BB58692B65382BF4A8B
```

Figure 10. Generate Public Key in Android

By using the key pair that generates from the server, this Android application can decrypt and get the same plain text as Figure 9 and Figure 10.

4.5. Android Volley Communication

In this implementation, Android volley is used for communication between mobile devices and servers. Volley is an HTTP library that makes networking for Android apps easier and faster. It integrates easily with any protocol and comes out of the box with support for raw strings, images, and JSON. In Android volley, internet permissions, data request, network setup, cache, singleton pattern and JSON variable should be determined so between the server and the mobile device can be connected to network and data can be received correctly.

5. Conclusion

Recently, payment transaction uses a credit card, debit card, etc. That method still doesn’t have a good security system. Therefore, in this paper, we proposed mobile payment system to use mutual authentication for a safer security. This mobile payment system combines QR Code and also fingerprint to provide a better level of security. Because each of our fingers has their unique pattern, we can bring the additional layer of protection for authentication by making a sequence of which finger should be read first, second, and so on. The adoption of QR Code ensures the legibility, integrity, and confidentiality of the transmitted information. It also lays a good foundation for the overall security of the system. In this implementation, RSA is used as the encryption and decryption algorithm. Public Key Infrastructure (PKI) is used as the key distribution mechanism. The third party is involved in generating the key pair of the public and the private key. All these security methods are combined to make mobile payment system more secure than ever.

Acknowledgements

This work was supported by a Research Grant of Pukyong National University(2016Year).

References

[1] William Stallings, “Cryptography and Network Security”, Pearson, United States of America.

[2] Shuang Zheng and Linfan Zhang, “Enhancing QR Code Security”, School of Health and Society, Departement Design and Computer Science, Swedan, (**2015)**.

[3] Ariana Tulus Purnomo, Yudi Satria Gondokaryono, Chang-Soo Kim, Ilkyeun Ra, “A Study on QR Code and Fingerprint Sequence for Securing Mobile Payment System”, International Conference on Future Information & Communication Engineering, (**2016)**, pp. 33-35.

[4] Ariana Tulus Purnomo, Vincentius Timothy, Yudi Satria Gondokaryono, Chang-Soo Kim, “An Approach in Mobile Payment Security using QR Code and ID-Based Encryption through Public Key Infrastructure (PKI)”, Conference on Information Security and Cryptography, (**2016)**.

[5] Xiangpeng Fu, Kaiying Feng, Changzhong Wang, and Junxing Zhang, “Improving Fingerprint Based Access Control System Using Quick Response Code”, International Conference on Computational Intelligence and Virtual Environments for Measurement Systems and Applications, **(2015)**.

[6] Michael Gordon, Suresh Sankaranarayanan, “Biometric Security Mechanism In Mobile Payments”, Seventh International Conference on Wireless and Optical Communications Networks, **(2010)**.

[7] Kinjal H. Pandya and Hiren J. Galiyawala, “A Survey on QR Codes: in contex of Research and Application”, International Journal of Emerging Technology and Advanced Engineering, **(2014).**

[8] Taolin MA, Jun QIAN, Yufei TIAN, Huixu ZHANG and Xinglong HU, “The Design and Implementation of an Innovative Mobile Payment System Based on QR Bar Code”, International Conference on Network and Information System for Computers, **(2015)**, pp. 435-440**.**

[9] Prime Number Hide-and-Seek: How the RSA Cipher Works. (n.d.). accessed from muppetlabs.com: http://www.muppetlabs.com/~breadbox/txt/rsa.html#11

Authors

Ariana Tulus Purnomo has received her Bachelor Degree in Electrical Engineering from Institut Teknologi Bandung, Indonesia, in August 2015. She is currently pursuing Master Dual-Degree both in Electrical Engineering especially in Information Security field from Institut Teknologi Bandung, Indonesia, and in IT Convergence and Application Engineering from Pukyong National University, South Korea. Her main research interests include mobile payment system, FPGA, and CNC Machine.

Yudi Satria Gondokaryono is an assistant professor in computer engineering and the Director of Information System and Technology Office of Institut Teknologi Bandung (ITB). His research interests include a high-performance system and computer security. He received his bachelor degree from Institut Teknologi Bandung in 1989, and an MS in Electrical Engineering (1997) and a Ph.D. in Electrical Engineering (2003) from New Mexico

State University. He is a member of the Technical Committee on Parallel Processing (TCPP) IEEE and member of IEEE and ACM.

Chang Soo Kim received a B.S Degree in Computer Science from Ulsan University, Korea, in 1979, and an M.S. degree in Computer Engineering and Ph.D. Degree in Computer Engineering from Chungang University, Korea, in 1984 and 1991 respectively. He has been a professor at the Department of IT Convergence and Application Engineering, Pukyong National University, Korea, since 1992. His research interests are operation system, LBS/GIS, WSN and urban disaster prevention system

Ilkyeun Ra holds a Ph.D. Degree in Computer and Information Science from Syracuse University in 2001, MS Degree in Computer Science from University of Colorado Boulder, and BS degree and MS degree in Computer Science from Sogang University. He was a Research Staff Member at the LG Information and Communications (Currently LG Telecom) Research Center. He joined the Department of Computer Science and Engineering at the University of Colorado Denver 2001. His main research interests include computer networks, developing adaptive distributed system software and high-speed communication system software to support High-Performance Distributed Computing Applications