Wireless Pers Commun (2017) 97:1–17
DOI 10.1007/s11277-017-4261-9

A Secure and Efficient Key Authentication using Bilinear Pairing for NFC Mobile Payment Service

Xinyi Chen[1] · Kyung Choi[2] · Kijoon Chae[1]

Published online: 22 September 2017

Abstract Near Field Communication (NFC) is widely used as a contactless communication technology in mobile phones for mobile payments. However, achieving payment security is challenging due to the authentication between the NFC-enabled mobile phone users and the merchants. Attackers can use the vulnerabilities of card payment transactions to compromise the NFC communication message and then transmit the wrong payment information to the communicators. An efficiency key authentication scheme is proposed to help NFC-enabled mobile payment communication using bilinear pairing. The proposed scheme can furnish a secure environment for NFC mobile payments by providing unlinkability and unforgeability functions to prevent attack scenarios.

Keywords NFC mobile payment · Bilinear pairing · ECC · Authentication

1 Introduction

Near Field Communication (NFC) is a new contactless technology that builds on the (13.56 MHz) RFID standard ISO/IEC 14443 and communicates within a short range to enable data exchange between devices at a distance of a few centimeters [1]. Embedded NFC technology in mobile phones for payment transactions is broadly used in current mobile payment systems based on contactless infrastructures. Additional security

✉ Kijoon Chae
kjchae@ewha.ac.kr

Xinyi Chen
chloexiny@nate.com

Kyung Choi
cbk0907@gmail.com

[1] Department of Computer Science and Engineering, Ewha Womans University, Seoul, Korea

[2] School of Information and Communication Engineering, Sungkyunkwan University, Seoul, Korea

functionality has just started to emerge in mobile phones. Instead of issuing physical contactless cards, an NFC-enabled mobile phone can act primarily as either a reader or a token, and mobile payment using NFC phones will become a reality due to consumer demand. For example, with mobile payments as services independent of mobile banking, the consumer using an NFC-enabled device in card emulation mode can pay for goods in front of a Point-of-Sale merchant machine. However, payment scenarios based on NFC technology can raise security challenges when considering the vulnerable communication protocol between two NFC devices in mobile payment services. Intruders or malicious users trying to issue bogus payments or impersonate legitimate users should be prevented from accessing services. Also, malicious payees impersonating legitimate payees should not be able to receive payments for services they do not provide, as described in [2, 3]. Another concern is that some external adversaries or payees could be able to develop a logic-linkable connection by intercepting the communication messages in an insecure transmission channel or by exploiting the payment system.

Establishing a secure channel between two NFC devices is clearly the best approach to protect against eavesdropping and any kind of data modification or fabrication attack for NFC-enabled mobile payment transactions. Due to the inherent protection of NFC against attacks, such as man-in-the-middle attacks, it is rather straightforward to setup a secure channel. A standard key agreement protocol, like the Diffie-Hellmann protocol based on RSA or Elliptic Curves, could be applied to establish a shared secret between two devices. However, it is not always an easy task considering the very limited capacities of NFC-enabled mobile phones and NFC-enabled computation environments. Traditional public-key cryptography requires heavy computation and a long execution time, and may not be a good solution in NFC-enabled mobile payment communication [4]. Elliptic Curve Cryptography (ECC) provides the most efficient memory utilization for constrained NFC communication environments by a wide margin over equivalent RSA-based solutions. One of the significant benefits of ECC is that it saves memory space, as it can provide equivalent crypto strengths with smaller key sizes (in bits) compared with other cryptographic techniques such as RSA. Since establishing a shared secret is difficult for an NFC-enabled communication environment, a "paired secret" is considered by using a bilinear pairing function combined with the ECC method to achieve both security and efficiency. Bilinear pairing gives rise to new mathematical problems that can be used as a base for secure cryptosystems. Let G_1 and G_2 be additive cyclic groups of order n. Let G_3 be a multiplicative cyclic group of order n. A bilinear pairing is an efficiently computable map ê: $G_1 \times G_2 \rightarrow G_3$ which satisfies the following three general properties:

- *Bilinearity*

  (a) $\hat{e}(aP_1, P_2) = \hat{e}(P_1, P_2)^a = \hat{e}(P_1, aP_2)$
  (b) $\hat{e}(aP_1, bP_2) = \hat{e}(P_1, P_2)^{ab}$
  (c) $\hat{e}(P_1 + P_2, Q) = \hat{e}(P_1, Q)\,\hat{e}(P_2, Q)$

- *Non-degeneracy* There exists $P \in G_1$, $Q \in G_2$ such that $\hat{e}(\mathrm{P, Q}) \neq \mathrm{I}_{G3}$, where I_{G3} is an identity element of G_3. Note that the map ê does not send all pairs in $G_1 \cdot G_2$ to the identity in G_3. If P is a generator of the group G_1, then $\hat{e}(P, P)$ is a generator of the group G_3.
- *Computability:* There must be an efficient algorithm, which can compute $\hat{e}(\mathrm{P, Q})$ for all $P, Q \in G_3$.

Our paper aims to offer an efficient key authentication scheme based on an NFC-enabled mobile payment service using bilinear pairing. Instead of using the traditional RSA public-key encryption/decryption method, the proposed scheme uses a lightweight ECC cryptography method based on the properties of the bilinear pairing and can achieve the same level of unlinkability and unforgeability as the traditional RSA method.

This paper is structured as follows. Section 2 gives an introduction to related works in security research in NFC and introduces bilinear pairing. Section 3 describes the proposed NFC-enabled mobile payment authentication scheme, and Sect. 4 gives a description of a simulation and operation test of the authentication scheme, including a summary of the security functionalities and computational time of the proposed NFC mobile payment authentication mechanism. An analysis to prove that the proposed scheme is more efficient and secure than previous approaches is given in Sects. 5 and 6 presents the conclusions.

2 Related Works

The distance of NFC is much shorter than the existing RF wireless communications. However, the attacks that could happen in traditional wireless environments can also happen in NFC-enabled mobile payment communication. Threats such as eavesdropping, corruption, insertion and man-in-the-middle-attacks can still disrupt NFC transactions, even though it only has a communication range of 10 cm. A more secure mechanism is needed for the authentication between two entities communicating in an NFC environment.

Currently, NFC security technology is defined by the NFC Forum, which defined a type of signature record in [5]. This development is based on the ECMA (Electronic Computer Manufacturers Association) standard. It focuses on finding a more flexible way to conduct financial transactions and information sharing for an individual consumers. H. C. Cheng and W. W. Liao [6] presented a key management and authentication scheme based on the RSA public key algorithm in NFC Read/Write mode. However, the authentication was based on a traditional public key and was not fast enough for NFC communication within a short period of short time. Moreover, their scheme did not verify the merchant in the mobile payment system, whereas the Point-of-Sale merchant cannot be assumed to as a legal seller. In 2011, E. Husni et al. [7] proposed a Tag-to-Tag NFC protocol that could realize mutual authentication for a consumer and merchant. However, the protocol used both a symmetric key and the consumer's password, which cannot meet the requirements of highly dynamic NFC mobile payments. ECC is by far the most efficient security solution for constrained NFC communication environments with mobile devices, as compared to equivalent RSA based solutions, and provides a 10-fold reduction in the storage overhead compared to RSA signatures and certificates (from about 1000 to 100 bytes) [8].

Compared with methods that use a symmetric key or pre-shared secret, pairing is more useful for exchanging security information for payments in NFC-enabled mobile communication, particularly in pairing-based ECC. Bilinear pairings on elliptic curves such as Weil pairing or Tate pairing have recently found positive applications in cryptography [9]. The modified Weil or Tate pairing can be used as a symmetric pairing. When a cryptographic protocol requires a symmetric pairing, a super singular curve with a distortion map should be chosen. When a cryptographic protocol requires an asymmetric pairing, then an elliptic curve should be chosen [10]. These methods are related to the discrete logarithm problem investigated for finite fields.

Bilinear pairing has already been used for secure communication in previous research. H. Du and Q. Yen [11] proposed an efficient and provably secure certificateless short signature scheme from bilinear pairings. They presented a certificateless signature (CLS) scheme that was proven to be secure in the random oracle model and required general cryptographic hash functions instead of the map-to-point hash function, which is inefficient. Chen et al. [12] proposed a novel e-cash system based on identity-based bilinear pairing to create an anonymity revocation function. They constructed an identity-based blind signature scheme, in which a bank can blindly sign a message containing a trustee-approved token that includes the user's identity. Liao and Hsiao [13] proposed a novel multi-server remote user authentication scheme using self-certified public keys for mobile clients, and the proposed scheme achieved mutual authentication and session key agreement. The scheme can withstand an offline dictionary attack, due to the security breach of mobile devices, and enhance the password change phase with the help of the registration server. The cryptography of the bilinear pairing method is based on ECC. This study uses a combination of this cryptography method and bilinear pairing to protect mobile payments, and does not add any signature or certificate to the communication. An efficient way is proposed to encrypt the messages to be transmitted and authenticate them by decrypting the received message between two targets based on normal ECC methods. By using ECC, with a similar communication mode, the computation in the proposed scheme is faster than RSA-based schemes and can achieve the same level of security with a smaller key size.

3 Proposed NFC Mobile Payment Mechanism

This section introduces the proposed efficient key authentication scheme based on NFC-enabled mobile payment services by using bilinear pairing.

3.1 System Architecture

The architecture of the mobile payment system which provides a secure environment for the proposed NFC mobile payment key authentication mechanism is shown in Fig. 1.

The mobile payment system is constructed with three entities: a Consumer, Merchant, and Bank. Moreover, it assumes that neither the Consumer nor the Merchant trust each other, and that neither of them have verification capabilities for the other. An insecure payment system would not be acceptable to either the merchants or customers. Therefore, in this paper, we designate the Bank as a trusted third party for the authentication of the payment between the Consumer and the Merchant. If the Consumer or Merchant wants a secure way to communicate, they should prove that they have the right identity for the authentication phases operated by the Bank. Before the payment is made, the authentication information belonging to the Consumer and the Merchant, by means of which the verification is achieved, should be sent to the Bank. After successful authentication, the Bank will send a message to tell both the Consumer and Merchant that the payment has been made securely and successfully. Four stages are included in the secure mobile payment authentication system. The first stage (1) in Fig. 1 shows the contactless communication between the Consumer and Merchant. The second stage (2) concludes when the Merchant generates the authentication message and forwards it to the Bank. The third stage (3) is composed of the authentication phases in which the Bank verifies the payment communication between the Merchant and Consumer. In the first stage, the Consumer

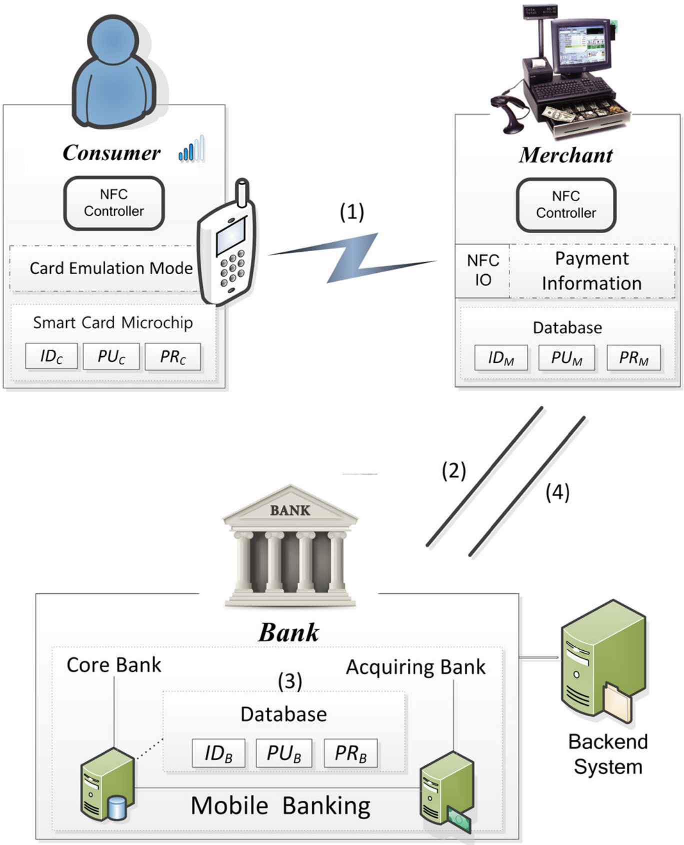

Fig. 1 The NFC-enabled mobile payment system

should send its authentication data to the Merchant. In the second stage, the Merchant will add its authentication data to that of the Consumer and then forward this combined information to the Bank. The third stage shows that the Bank receives these authentication data and performs the verification procedure for the received payment information. After that, a confirmation message will be sent from the Bank to the Merchant, as shown in part (4) of Fig. 1.

Finally, as an optional choice, a payment notification message will be sent to the Consumer to notify them of the success of the mobile payment as the fourth stage. The notations used throughout this paper are shown in Table 1.

3.2 Assumptions of the NFC-Enabled Mobile Payment Communication

Before introducing the proposed scheme, the assumptions made for the NFC-enabled mobile payment communication system should be mentioned:

1. A trust third party is responsible for generating Consumer's private key and public key pair. The Merchant's public key and private key is generated by the trust third party. The Bank B's public key and private key is also generated by the trust third party.

Table 1 Notations

Notation	Description
ID_C	Consumer C's identification
ID_M	Merchant M's identification
ID_B	Bank B's identification
PU_C, PR_C	Consumer C's public key and private key
PU_M, PR_M	Merchant M's public key and private key
PU_B, PR_B	Bank B's public key and private key
OI	Ordering Information, contains the ordering number and the products' price, etc.
G	Elliptic curve based point
R	Random integer generated by Merchant M
Q	Random integer generated by Merchant M
N	Random integer generated by Consumer C
C_{Auth}	Consumer C's authentication information
M_{Auth}	Merchant M's authentication information
MC^{+}	Combination of C_{Auth} and M_{Auth}
M_k	One of pairing messages, contains PR_M and r
K_M	Encryption/decryption key generated by Merchant M
E_{KM}, D_{KM}	Encryption/decryption function with K_M
M_{PAY}	Mobile payment information which encrypted by K_M
T_S	Timestamp of Merchant M sending M_{PAY}
T_B	Timestamp of Bank B sending confirmation message
$H(\cdot)$	One-way hash function

2. Each Consumer's NFC phone stores its ECC public key (PU_C) and the private key (PR_C): $PU_C = PR_C \cdot G$.
3. Each Merchant stores its ECC public key (PU_M) and the private key (PR_M): $PU_M = PR_M \cdot \mathrm{G}$.
4. The Bank holds its own public key (PU_B) with the private key (PR_B): $PU_B = PR_B \cdot \mathrm{G}$.
5. Each Consumer and each Merchant know the Bank's ID_B and the public key PU_B.
6. The Bank stores the Consumer's ID_C with its public key PU_C and Merchant's ID_M with its public key PU_M.
7. The Bank, Consumer and Merchant have agreed to a base point Q as well as a hash function.

3.3 Proposed Mechanism

The proposed NFC-enabled mechanism can be seen as a cryptographic protocol that requires an asymmetric pairing based on random oracle using the ECC secure method. A description of this mechanism will follow the four stages mentioned below, which can be separated into eleven steps. The process is shown in Fig. 2.

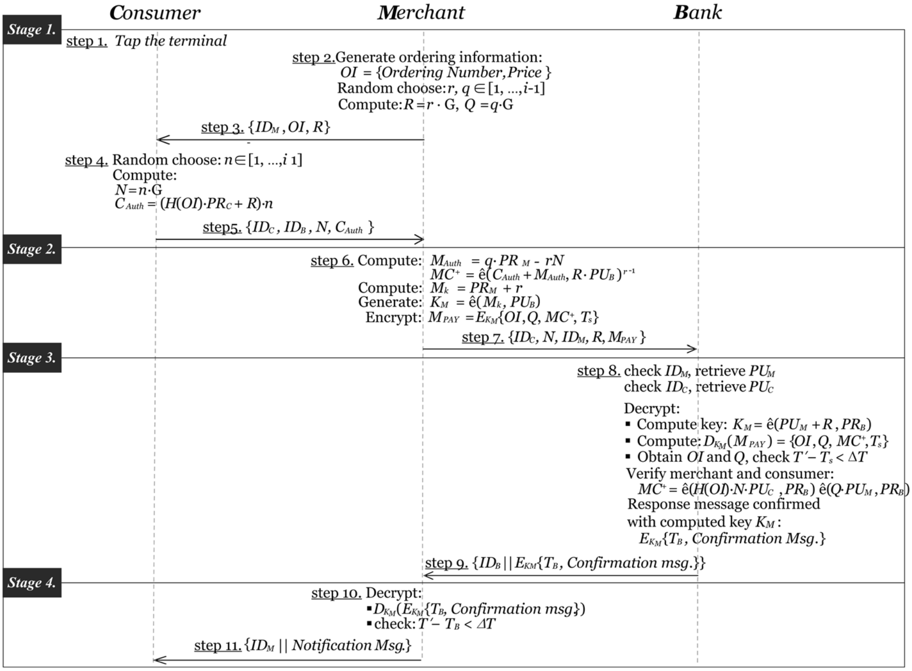
Consumer
Merchant
Bank
Stage 1.
step 1. Tap the terminal
step 2. Generate ordering information:
OI = {Ordering Number, Price }
Random choose: r, q ∈ [1, ...,i-1]
Compute: R = r · G, Q = q·G
step 3. {ID_M , OI, R}
step 4. Random choose: n ∈ [1, ...,i 1]
Compute:
N=n·G
C_Auth = (H(OI)·PR_C + R)·n
step5. {ID_C , ID_B , N, C_Auth }
Stage 2.
step 6. Compute: M_Auth = q·PR_M - rN
MC⁺ = ê(C_Auth + M_Auth, R·PU_B)^r-1
Compute: M_k = PR_M + r
Generate: K_M = ê(M_k, PU_B)
Encrypt: M_PAY = E_KM{OI, Q, MC⁺, T_s}
step 7. {ID_C, N, ID_M, R, M_PAY }
Stage 3.
step 8. check ID_M, retrieve PU_M
check ID_C, retrieve PU_C
Decrypt:
▪ Compute key: K_M = ê(PU_M + R , PR_B)
▪ Compute: D_KM(M_PAY) = {OI, Q, MC⁺, T_s}
▪ Obtain OI and Q, check T′ − T_s < ΔT
Verify merchant and consumer:
MC⁺ = ê(H(OI)·N·PU_C , PR_B) ê(Q·PU_M , PR_B)
Response message confirmed
with computed key K_M :
E_KM{T_B , Confirmation Msg.}
step 9. {ID_B || E_KM{T_B , Confirmation msg.}}
Stage 4.
step 10. Decrypt:
▪ D_KM(E_KM{T_B, Confirmation msg})
▪ check: T′ − T_B < ΔT
step 11. {ID_M || Notification Msg.}

Fig. 2 Proposed NFC mobile payment authentication mechanism

Stage 1 *The Consumer makes and sends a contactless mobile payment with its executed authentication information* (*to meet the security requirement*) *to the Merchant. The phases* (*from step* 1 *to step* 5) *are as follows*:

Step 1 Consumer *C*: The payment is made with the NFC phone by bringing it close to the Merchant *M*'s POS machine.

Step 2 Merchant *M*: The Ordering Information ($OI = Ordering\ Number,\ Price$) is generated for the *C*'s payments. Then it selects a random integer number *r*, *q* from the field [1, …, $i - 1$], and uses the randomly generated point G to calculate the value $R = r \cdot \mathrm{G}$, $Q = q \cdot \mathrm{G}$.

Step 3 Merchant $M \rightarrow$ Consumer *C*: The payment information *OI* is sent with the *M*'s ID_M and random number *R* to *C*.

Step 4 Consumer *C*: An integer *n* is randomly chosen and the randomly generated point G is used to calculate $N = n \cdot \mathrm{G}$. Next, *C* calculates its authentication information C_{Auth} as the proof and later sends it to Bank *B* through *M* for the purpose of implementing the payment phase successfully:

$$C_{Auth} = (H(OI) \cdot PR_C + R) \cdot n \tag{1}$$

To prevent redundancy, Consumer *C* should add its private key PR_C to $H(OI)$.

Step 5 Consumer $C \rightarrow$ Merchant *M*: Consumer C sends its ID_C with its account Bank *B*'s ID_B and random number *N* with the authentication C_{Auth} to *M*.

Stage 2 *After receiving the payment requirement information from the Consumer, the Merchant generates and adds its authentication data to the received Consumer authentication data and forwards the encrypted payment information to the Bank with the ordering information. Step* 6 *and step* 7 *show the authentication information generation phases and transmission phases, respectively*:

Step 6 Merchant M: The following calculation is performed if M receives the message sent by C:

$$M_{Auth} = q \cdot PR_M - rN \tag{2}$$

$$MC^{+} = \hat{e}(C_{Auth} + M_{Auth}, R \cdot PU_B)^{r-1} \tag{3}$$

M_{Auth} is the authentication information constructed by M, which will be verified by Bank B later. Next, it generates the encryption key K_M and uses it to encrypt the payment message $\{OI, Q, MC^{+}, T_s\}$ as M_{PAY}. The calculation is as follows:

$$M_k = PR_M + r \tag{4}$$

$$K_M = \hat{e}(M_k, PU_B) \tag{5}$$

$$M_{PAY} = E_{KM}\{OI, Q, MC^{+}, T_s\} \tag{6}$$

Step 7 Merchant M → Bank B: Merchant M forwards C's authentication information C_{Auth} after adding its own identification $\{ID_C, N, ID_M, R, MC^{+}, M_{PAY}\}$, which together serve to validate their communication, together with the encrypted message, to the Bank B.

Stage 3 The *Bank receives the authentication information sent by the Merchant and executes the bilinear pairing-based exponentiation procedure to confirm the validity of the proof contained in the authentication information. A payment confirmation message will be sent back to the Merchant if the verification succeeds. The phases can be described as below:*

Step 8 Bank B: After receiving the message from M, B can obtain C and M's relevant public keys by retrieving ID_C and ID_M and checking them. Next, the plain text $\{OI, Q, T_s\}$ is obtained from the decrypted message.

$$K_M = \hat{e}(PU_M + R, PR_B) \tag{7}$$

$$D_{KM}(M_{PAY}) = \{OI, Q, MC^{+}, T_s\} \tag{8}$$

$$\text{check} : T' - T_s < \Delta T \tag{9}$$

After obtaining the message $\{OI, Q\}$ from the decrypted M_{PAY} message, B uses $\{OI, Q\}$ and the received message $\{N, R, MC^{+}\}$ to verify C and M:

$$MC^{+} = \hat{e}(H(OI) \cdot N \cdot PU_C, PR_B)\hat{e}(Q \cdot PU_M, PR_B) \tag{10}$$

The computations to verify whether MC^{+} is equal to the result of pairing are as follows:

$$
\begin{aligned}
MC^{+} &= \hat{e}(C_{Auth} + M_{Auth}, R{\cdot}PU_B)^{r-1} \qquad (3)\\
&= \hat{e}((H(OI){\cdot}PR_C + R){\cdot}n + M_{Auth}, R{\cdot}PU_B)^{r-1}\\
&= \hat{e}((H(OI){\cdot}PR_C + R){\cdot}n + q{\cdot}PR_M - rN, R{\cdot}PU_B)^{r-1}\\
&= \hat{e}(H(OI){\cdot}PR_C{\cdot}n + \ R{\cdot}n + q{\cdot}PR_M - rN, R{\cdot}PU_B)^{r-1}\\
&= \hat{e}(H(OI){\cdot}PR_C{\cdot}n + \ rG{\cdot}n + q{\cdot}PR_M - r{\cdot}nG, R{\cdot}PU_B)^{r-1}\\
&= \hat{e}(H(OI){\cdot}PR_C{\cdot}n + q{\cdot}PR_M, R{\cdot}PU_B)^{r-1}\\
&= \hat{e}\big(H(OI){\cdot}PR_C{\cdot}n{\cdot}r^{-1} + q{\cdot}PR_M{\cdot}r^{-1}, R{\cdot}PU_B\big)\\
&= \hat{e}\big(H(OI){\cdot}PR_C{\cdot}n{\cdot}r^{-1}{\cdot}R + q{\cdot}PR_M{\cdot}r^{-1}{\cdot}R, PU_B\big)\\
&= \hat{e}\big(H(OI){\cdot}PR_C{\cdot}n{\cdot}r^{-1}{\cdot}rG + q{\cdot}PR_M{\cdot}r^{-1}{\cdot}rG, PU_B\big)\\
&= \hat{e}(H(OI){\cdot}PR_C{\cdot}n{\cdot}G + q{\cdot}PR_M{\cdot}G, PU_B)\\
&= \hat{e}(H(OI){\cdot}PR_C{\cdot}N + PR_M{\cdot}Q, PU_B)\\
&= \hat{e}(H(OI){\cdot}PR_C{\cdot}N + PR_M{\cdot}Q, PR_B{\cdot}G)\\
&= \hat{e}(H(OI){\cdot}PR_C{\cdot}G{\cdot}N + PR_M{\cdot}Q{\cdot}G, PR_B)\\
&= \hat{e}(H(OI){\cdot}PU_C{\cdot}N + Q{\cdot}PU_M, PR_B)\\
&= \hat{e}(H(OI){\cdot}N{\cdot}PU_C, PR_B)\hat{e}(Q{\cdot}PU_M, PR_B) \qquad (10)
\end{aligned}
$$

The result of this proof procedure is:

$$
\begin{aligned}
MC^{+} &= \hat{e}(C_{Auth} + M_{Auth}, R \cdot PU_B)^{r-1}\\
&= \hat{e}(H(OI) \cdot N \cdot PU_C, PR_B)\hat{e}(Q \cdot PU_M, PR_B)
\end{aligned}
$$

If B obtains the above result, then the verification phases are considered to have succeeded. B then transmits an encrypted message that contains a confirmation message ({*Confirmation Msg.*}) as an announcement of the successful authentication to M. The transmission message will be added to the payment confirmation time T_B encrypted by the key K_M, which was calculated before.

$$E_{KM}\{T_B, Confirmation\,Msg.\} \tag{11}$$

Step 9 Bank B → Merchant M: B sends its ID_B integrated with the encrypted message $\{ID_B \parallel E_{KM}\{T_B,$ *Confirmation Msg.*$\}\}$ to M.

Stage 4 *The payment communication is accomplished by the Merchant sending its notification information to the Consumer, as described in step* 10 *and step* 11:

Step 10 *Merchant M*: After having received the message sent by B, M decrypts the message using key K_M and checks T_B.

$$D_{KM}(E_{KM}\{T_B, Confirmation\,Msg.\}) \tag{12}$$

$$check : T' - T_B < \Delta T \tag{13}$$

Step 11 *Merchant M* → Consumer C: After M checks the payment confirmed message, a notice message $\{ID_M \parallel$ *Notification Msg.*$\}$ will be sent to C.

This paper proposes that only Bank B has the right and ability to do the authentication process. Therefore, the successful payment phases will be done when Merchant M receives the confirmation message sent by the Bank. The last step of the notification information transmission procedure is not necessary only if the Consumer *C* requires confirmation of the payment service on the first occasion.

4 Functionalities and Cost Analysis

This section summarizes the security functionalities and total processing time of the proposed NFC mobile payment authentication mechanism.

4.1 Security Functionalities

Unlinkability means that when two messages are generated by the same Consumer, the connectivity between the two data should not be identifiable. Chen et al. doesn't provide a check of trust third party for payment information in micropayment protocol. Moreover, even if the merchant transmit a false payment amount to the third party, the third party can not verify a correctness of the payment amount in case of micropayment protocol. Unforgeability means that the transmission message should not be able to be faked by a dishonest Consumer or Merchant during communication. Table 2 shows a comparison of the security functionalities between the related schemes. It shows that our scheme could provide the same security functionalities as those obtained using the RSA scheme [6] [7].

4.2 Cost Analysis

Nowadays, it is well-known that most mobile devices (especially mobile phones) have enough energy resources and computing capability. Hence, a total processing time is more important issues than the power consumption in NFC-enabled mobile payment communication. When the Consumer contact the NFC-enabled mobile phone for payment, the processing time should be finish as soon as possible. The time cost of the computational or communication steps include the parameter generation (ordering information, random number, encryption key in this paper), verification phases, and waiting time (including round trip delay). The cost analysis shown in Fig. 3 includes an estimation of the computation time and communication time.

The computation time in the proposed scheme depends on the algorithm used to provide the cryptography services, such as the bilinear pairing operations and encryption/decryption based on ECC. The four stages mentioned in section III have time costs of 114 *ms*, 123 *ms*, 227 *ms,* and 56 *ms,* respectively. The first stage follows the normal contactless communication and only required a small computation with the ordering information. Therefore, it has similar communication and computation times to those of a normal NFC-enabled device. Without the security function, the connection between two NFC devices is established almost at once, requiring less than 0.1 s. For stages 2 and 3, the addition of the security functionalities increases the computation times to 123 ms and 227 ms, respectively. However, the waiting time for the mobile communication is somewhat long, since

Table 2 Security functionalities comparison

Functionalities	H. C. Chen et al. [6]	E. Husni et al. [7]	Proposed scheme
Unlinkability	Yes	No	Yes
Unforgeability	Yes	Yes	Yes

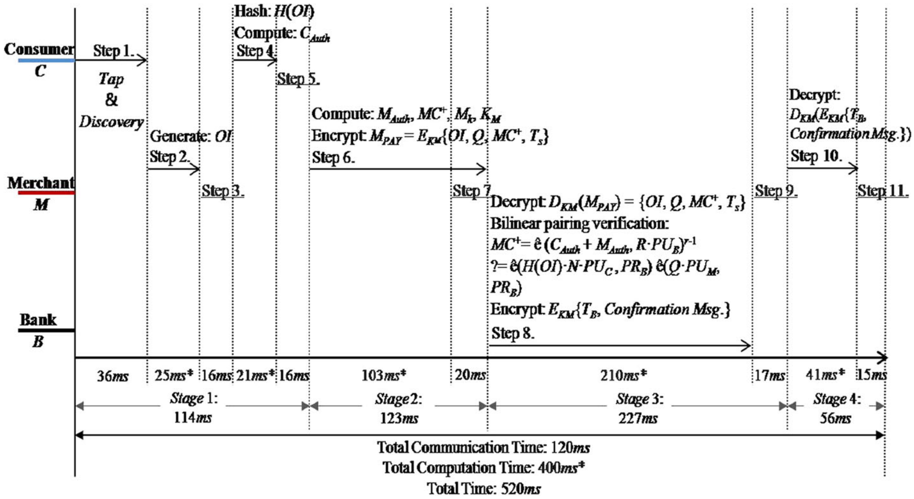
Consumer
C
Merchant
M
Bank
B
Step 1.
Tap & Discovery
Generate: OI
Step 2.
Step 3.
Hash: $H(OI)$
Compute: C_{Auth}
Step 4.
Step 5.
Compute: M_{Auth}, MC^{+}, M_{k}, K_{M}
Encrypt $M_{PAY} = E_{KM}\{OI, Q, MC^{+}, T_{s}\}$
Step 6.
Step 7.
Decrypt $D_{KM}(M_{PAY}) = \{OI, Q, MC^{+}, T_{s}\}$
Bilinear pairing verification:
$MC^{+} = \hat{e}(C_{Auth} + M_{Auth}, R \cdot PU_{B})^{r-1}$
$?= \hat{e}(H(OI) \cdot N \cdot PU_{C}, PR_{B})\, \hat{e}(Q \cdot PU_{M}, PR_{B})$
Encrypt $E_{KM}\{T_{B}, Confirmation\ Msg.\}$
Step 8.
Step 9.
Decrypt:
$D_{KM}(E_{KM}\{T_{B}, Confirmation\ Msg.\})$
Step 10.
Step 11.
36ms
25ms*
16ms
21ms*
16ms
103ms*
20ms
210ms*
17ms
41ms*
15ms
Stage 1: 114ms
Stage 2: 123ms
Stage 3: 227ms
Stage 4: 56ms
Total Communication Time: 120ms
Total Computation Time: 400ms*
Total Time: 520ms

Fig. 3 Estimated time required for the process of the proposed NFC mobile payment authentication mechanism

the total time is 520 ms, however this represents only about 0.5 s of waiting. The last stage has an additional service for the Consumer only if he or she chooses the notification service for payment. Otherwise, the last 15 ms can be omitted and the total time of the last stage is 56 ms. The total communication time is 120 ms and the total computation time is 400 ms, as shown in Fig. 3. Even though the proposed scheme uses more cryptography operations than the other schemes, this does not significantly affect the performance, since it operates efficiently within a short time. Table 3 summarizes the communication and computation time costs in each stage along with the performance of its steps.

Cheng et al. [6]. only performed a key obtaining process which would be finished in about 2 s. However, our total payment processing time is 520 ms including key generation and key authentication. Moreover, the total payment process time includes the computation time and the communication time from the Consumer through the Merchant to the Bank.

Table 3 Cost for each stage with its steps

Stage	Time	Step	Time
Stage 1	114 ms	Step 1	36 ms
		Step 2	25 ms
		Step 3	16 ms
		Step 4	21 ms
		Step 5	16 ms
Stage 2	123 ms	Step 6	103 ms
		Step 7	20 ms
Stage 3	227 ms	Step 8	210 ms
		Step 9	17 ms
Stage 4	56 ms	Step 10	41 ms
		Step 11	15 ms
Total	520 ms	11 steps	520 ms

5 Security Analysis

This section presents a security analysis of the NFC mobile payment key authentication mechanism, focusing on a hybrid of some well-known attacks such as eavesdropping. Definitions will be given combined with these attacks, and formal proofs of the correctness, unlinkability, and unforgeability properties of the proposed mechanism are presented. The formation of the proposed definitions and theorems and the attack model contribution are based on previous studies [14–16].

5.1 Correctness

This section will prove the correctness of the pairing computations when Bank B verifies that the formulation (3) can pair with Eq. (10) in step 8.

Theorem 1 *The proposed NFC mobile payment authentication mechanism satisfies the requirement of correctness.*

Proof According to Eqs. (1) and (2):

$$C_{Auth} = (H(OI) \cdot PR_C + R) \cdot n \tag{1}$$

$$M_{Auth} = q \cdot PR_M - rN \tag{2}$$

MC^+ can be represented as formulation (3):

$$MC^+ = \hat{\mathrm{e}}(C_{Auth} + M_{Auth}, R \cdot PU_B)^{r-1} \tag{3}$$

When the extension of Eqs. (1) and (2) are substituted into (3), MC^+ can be extended as:

$$MC^+ = \hat{\mathrm{e}}((H(OI) \cdot PR_C + R) \cdot n + q \cdot PR_M - rN, R \cdot PU_B)^{r-1}$$

Depending on the attribute ((a) $\hat{\mathrm{e}}(aP_1, P_2) = \hat{\mathrm{e}}(P_1, P_2)^a = \hat{e}(P_1, aP_2)$) of the bilinear pairing described in Sect. 1, the equation of MC^+ can be rewritten as:

$$MC^+ = \hat{\mathrm{e}}\big(H(OI) \cdot PR_C \cdot n \cdot r^{-1} + q \cdot PR_M \cdot r^{-1}, R \cdot PU_B\big)$$

It also can be seen that the equation depends on attribute (a) of the bilinear pairing:

$$MC^+ = \hat{\mathrm{e}}\big(H(OI) \cdot PR_C \cdot n \cdot r^{-1} \cdot R + q \cdot PR_M \cdot r^{-1} \cdot R, PU_B\big)$$

After several computations, the formulation of $MC^+ = \hat{\mathrm{e}}\ (H(OI) \cdot PU_C \cdot N + Q \cdot PU_M, PR_B)$ can be obtained.

$$\begin{aligned} MC^+ &= \hat{\mathrm{e}}\big(H(OI) \cdot PR_C \cdot n \cdot r^{-1} \cdot R + q \cdot PR_M \cdot r^{-1} \cdot R, PU_B\big) \\ &= \hat{\mathrm{e}}\big(H(OI) \cdot PR_C \cdot n \cdot r^{-1} \cdot rG + q \cdot PR_M \cdot r^{-1} \cdot rG, PU_B\big) \\ &= \hat{\mathrm{e}}(H(OI) \cdot PR_C \cdot n \cdot G + q \cdot PR_M \cdot G, PU_B) \\ &= \hat{\mathrm{e}}(H(OI) \cdot PR_C \cdot N + PR_M \cdot Q, PU_B) \end{aligned}$$

Since the public key PU_B of Bank B is made by its private key PR_B multiplied by the ECC public point G, and Consumer C ($PU_C = PR_C \cdot$ G) and Merchant M ($PU_M = PR_M \cdot$ G) have the same situation, the above formulation continues as:

$$\begin{aligned} MC^+ &= \hat{e}(H(OI) \cdot PR_C \cdot N + PR_M \cdot Q, PR_B \cdot G) \\ &= \hat{e}(H(OI) \cdot PR_C \cdot G \cdot N + PR_M \cdot Q \cdot G, PR_B) \\ &= \hat{e}(H(OI) \cdot PU_C \cdot N + Q \cdot PU_M, PR_B) \end{aligned}$$

Finally, using the last attribute of bilinear pairing ((*c*) $\hat{e}$ $(P_1 + P_2, Q) = \hat{e}$ (P_1, Q) $\hat{e}$ (P_2, Q)), the computation result is:

$$MC^+ = \hat{e}(H(OI) \cdot N \cdot PU_C, PR_B)\hat{e}(Q \cdot PU_M, PR_B) \tag{10}$$

The final computation result is the same as the extension formation of Eq. (3), and also shows correctness in that:

$$\begin{aligned} MC^+ &= \hat{e}(H(OI) \cdot N \cdot PU_C, PR_B)\hat{e}(Q \cdot PU_M, PR_B) \\ &= \hat{e}(C_{Auth} + M_{Auth}, R \cdot PU_B)^{r^{-1}} \end{aligned}$$

5.2 Unlinkability

This section will prove that the requirement of unlinkability is satisfied, in that when two messages are generated by the same Consumer, the connectivity between the two data should not be identifiable. The proposed NFC mobile payment authentication mechanism cannot be hacked by an attacker with any linkable information. To prove the unlinkability for the proposed NFC mobile payment authentication mechanism, we define two types of security, Type I and Type-II, against two types of adversaries, A_1 and A_2, respectively.

Adversary A_1 models a malicious adversary that compromises the messages transmitted on the special NFC communication channel between the Consumer and the Merchant. Adversary A_2 models a malicious adversary that compromises the wired communication channel between the Merchant and the Bank. There are four opportunities for the adversaries to attack:

Merchant M → *Consumer C*: The payment information *OI* is sent with the *M*'s ID_M and random number *R* to *C*.
Consumer C → *Merchant M*: Consumer C sends its ID_C with its account Bank *B*'s ID_B and random number *N* with the authentication C_{Auth} to *M*.
Merchant M → *Bank B*: Merchant M forwards *C*'s authentication information C_{Auth} and add its identification $\{ID_C, N, ID_M, R, MC^+, M_{PAY}\}$ which constitute proof of their communication, together with the encrypted message to the Bank *B*.
Bank B → *Merchant M*: Bnak B sends *B*'s ID_B with the encrypted message $\{ID_B \| E_{KM}\{T_B, \textit{Confirmation Msg.}\}\}$ to *M*.

Definition 1 If adversary A_1 eavesdrops on the communication channel between the Merchant and Consumer, then the $\{ID_M, OI, R\}$ sent by the Merchant to a passive Consumer's NFC phone can be captured at the very beginning of the transmission. Next, A_1 captures the $\{ID_C, ID_B, N, C_{Auth}\}$ message sent by the Consumer and obtains the value *R* and C_{Auth} from the transmission, then uses these values to do the offline guessing analysis.

Game 1 A_1 obtains the correct result from the guessing attack and can move to the next step of abstracting the Consumer's private key PR_C, which is what A_1 really wants from the captured message. Moreover, A_1 can use the guessed private key PR_C to make false information, which is very harmful for the key's owner. It can be seen that A_1 wins Game 1.

Game 2 A_2 obtains the correct result from the guessing attack and can move to the next step of abstracting the Merchant's private key PR_M, which is what A_1 really wants from the captured message. Moreover, A_2 can use the guessed private key PR_M to make false information, which is very harmful for the key's owner. It can be seen that A_2 wins Game 2.

Theorem 2 *The proposed NFC mobile payment authentication mechanism is secure against eavesdropping over the NFC communication channel.*

Proof Assume that the message sent by the Merchant $\{ID_M, OI, R\}$ and the message $\{ID_C, ID_B, N, C_{Auth}\}$ sent by the Consumer are captured by the adversary A_1. Then, there is a constructed solution that can help to break the attack assumption with unknown random quantities n. After receiving the message $\{ID_C, ID_B, N, C_{Auth}\}$ sent by the Consumer, A_1 will perform the following computation:

$$C_{Auth}? = C'_{Auth} = (H(OI) \cdot PR_C + R) \cdot n$$

Although A_1 can obtain the useful data OI and R for the C_{Auth} guessing computation, there is another unknown number n besides the private key PR_C. It is not possible to compute an equation with two unknown numbers at the same time. Even if A_1 guesses both of them correctly, the time required to do so is long enough for the Consumer to update the key or change the key for the next computation. Therefore, it can be said that the proposed NFC mobile payment authentication mechanism can defend against the eavesdropping attack, guessing attack and relay attack, and is secure with regard to unlinkability. Relay attacks exploit that a contactless token within communication range is in close proximity, by placing a proxy-token in range of a contactless reader and relaying communication over a greater distance to a proxy-reader communication with the authentic token. However, even if the message from the Merchant to the Consumer is relayed, it is no use of relaying the message because the attacker doesn't know the Consumer's private key and random quantities n. Therefore, the attacker is unable to make an appropriate response. Even if the message from the Consumer to the Merchant is relayed, it is also no use because the message is made by the specific Merchant.

5.3 Unforgeability

In the NFC-based mobile payment system, unforgeability means that the transmission message should not be able to be faked by a dishonest Consumer or Merchant during communication. The proposed NFC mobile payment authentication mechanism is existentially unforgeable against adaptive chosen message attacks under the assumption that the attacker cannot obtain either the Consumer or the Merchant's private key.

This section defines adversary A_3 models, with a dishonest Consumer or a dishonest Merchant who tries to control the real payment data, which can be used to cheat the trusted third party Bank.

Definition 2 Adversary A_3 models a dishonest Consumer who does not compute the real ordering information $OI = \{\text{Ordering Number, Price}\}$ with the correct payment. For example, a dishonest Consumer always wants to pay less than the real price. Therefore, the ordering information can be changed from the received *OI* in order to pay less. A_3 can also model a dishonest Merchant who does not input the real ordering information *OI* with the correct payment. For example, a dishonest Merchant always wants a Consumer to pay more than the real price. Therefore, the ordering information can be changed from the received *OI* in order to receive a higher payment.

Game 3 Adversary A_3, who is a dishonest Consumer, does not compute the real ordering information *OI*, but allows a much lower price than the real price to be transmitted in the received *OI* from Merchant and then computes the hash of the fake ordering information by normal hashing. If A_3 acts as the Consumer and passes the authentication and obtains the payment confirmation message, then A_3 wins Game 3.

Game 4 Adversary A_3, who is a dishonest Merchant, tries to control the real payment data, then makes a fake price and sends the fake ordering information without showing the fake information to the Consumer. If A_3 acts as the Merchant, passes the authentication, and obtains the payment confirmation message, then A_3 wins Game 4.

Theorem 3 *The proposed NFC mobile payment authentication mechanism is secure against a dishonest Consumer or Merchant's fake information.*

Proof Assume that A_3, who acts as a dishonest Consumer or Merchant, computes fake ordering information *OI* and transmits the fake *OI* to the Bank. Depending on the design of the proposed NFC mobile payment authentication mechanism, the Bank will first abstract the *OI* supported by the honest/dishonest Merchant by decrypting the received message. Next, the *OI'* received from the honest/dishonest Merchant is compared with the *OI″* sent by the honest/dishonest Consumer:

$$H(OI')? = H(OI'')$$

Finally, both the Consumer and Merchant's authentication information is verified by using bilinear pairing. It will be shown that if the two *OIs* are not the same, the pairing fails. Therefore, the proposed NFC mobile payment authentication mechanism is sufficiently secure against dishonest Consumers or Merchants that make fake payment information.

As a result, the proposed mechanism provides the authentication, key authentication and prevents the data modification and fabrication attack.

6 Conclusion

NFC technology is now available on mobile phones and its use has risen sharply. Transaction technology based on NFC mobile payments is ready, but problems concerning the device and terminal availability and some security-related issues persist. This paper proposed an efficiency key authentication scheme based on the NFC-enabled mobile payment service using bilinear pairing. The proposed scheme uses a lightweight ECC based on the properties of the bilinear pairing instead of using the traditional heavy RSA public-key cryptography method. Using ECC is more efficient and can achieve the same security with a smaller key size than RSA. Using the properties of bilinear pairing is more feasible and convenient for manual authentication in mobile payment communication.

Acknowledgements This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIP) (No. 2016R1A2B4015899).

References

1. Technical Specification (2008). Essentials for successful NFC ecosystem. *NFC Forum.*
2. Francis, L., Hancke, G., Mayes, K., & Markantonakis, K. A security framework model with communication protocol translator interface for enhancing NFC transactions. In *Telecommunications (AICT), 2010 sixth advanced international conference on, 2010* (pp. 452–461).
3. Mulliner, C. Vulnerability analysis and attacks on NFC-enabled mobile phones. In *Availability, reliability and security, 2009. ARES'09. International conference on, 2009* (pp. 695–700).
4. Blass, E.-O., Kurmus, A., Molva, R., & Strufe, T. (2013). PSP: Private and secure payment with RFID. *Computer Communications, 36*(4), 468–480.
5. Technical Specification (2010). Signature Record Type Definition. *NFC Forum.*
6. Cheng, H.-C., Liao, W.-W., Chi, T.-Y., & Wei, S.-Y. A secure and practical key management mechanism for NFC read-write mode. In *Advanced communication technology (ICACT), 2011 13th International conference on, 2011* (pp. 1095–1011).
7. Husni, E., Kuspriyanto, K., Basjaruddin, N., Purboyo, T., Purwantoro, S., & Ubaya, H. Efficient tag-to-tag near field communication (NFC) protocol for secure mobile payment. In *Instrumentation, communications, information technology, and biomedical engineering (ICICI-BME), 2011 2nd international conference on, 2011* (pp. 97–101).
8. Rosati, T., & Zaverucha, G. Elliptic curve certificates and signatures for nfc signature records. In *2011*: Citeseer.
9. Dutta, R., Barua, R., & Sarkar, P. (2004). Pairing-based cryptography: A survey. *Cryptology ePrint Archive, Report 2004/064*. http://eprint.iacr.org/2004/064.
10. Freeman, D., Scott, M., & Teske, E. (2010). A taxonomy of pairing-friendly elliptic curves. *Journal of Cryptology, 23*(2), 224–280.
11. Du, H., & Wen, Q. (2009). Efficient and provably-secure certificateless short signature scheme from bilinear pairings. *Computer Standards & Interfaces, 31*(2), 390–394.
12. Chen, Y., Chou, J.-S., Sun, H.-M., & Cho, M.-H. (2011). A novel electronic cash system with trustee-based anonymity revocation from pairing. *Electronic Commerce Research and Applications, 10*(6), 673–682.
13. Liao, Y.-P., & Hsiao, C.-M. (2013). A novel multi-server remote user authentication scheme using self-certified public keys for mobile clients. *Future Generation Computer Systems, 29*(3), 886–900.
14. Hafizul Islam, S., & Biswas, G. (2013). Provably secure certificateless strong designated verifier signature scheme based on elliptic curve bilinear pairings. *Journal of King Saud University-Computer and Information Sciences, 25*(1), 51–61.
15. Xiong, H., Guan, Z., Chen, Z., & Li, F. (2013). An efficient certificateless aggregate signature with constant pairing computations. *Information Sciences, 219,* 225–235.
16. Fan, C.-I., Sun, W.-Z., & Huang, V. S.-M. (2010). Provably secure randomized blind signature scheme based on bilinear pairing. *Computers & Mathematics with Applications, 60*(2), 285–293.

Xinyi Chen was born in Shanghai, China in 1988. She received the B.S. degree in computer engineering from Kyungil University in 2011, an M.S. degree in computer science and engineering from Ewha Womans University in 2013. Her research interests include access control, user authentication, mobile security, and NFC.

Kyung Choi received the B.S. degree in computer science from Yonsei University in 1995, an M.S. degree in information and science from Ewha Womans University in 2008, and a Ph.D. degree in computer science and engineering from Ewha Womans University in 2014. She is currently working as a postdoctoral researcher at the school of information and communication engineering in Sungkyunkwan University, Seoul, Korea. Her research interests include home network security, sensor network security, smart grid security, and cloud computing.

Kijoon Chae received the B.S. degree in mathematics from Yonsei University in 1982, an M.S. degree in computer science from Syracuse University in 1984, and a Ph.D. degree in electrical and computer engineering from North Carolina State University in 1990. He is currently a professor of computer science and engineering at Ewha Womans University, Seoul, Korea. His research interests include network security, home network, sensor network, smart grid, content delivery network, network protocol design and performance evaluation.