AN IMPROVEMENT TO THE SET PROTOCOL BASED ON SIGNCRYPTION

Hassan M. Elkamchouchi¹, Eman F. Abu Elkhair² and Yasmine Abouelseoud³

¹ Elec. Eng. Dept, Fac. of Eng., Alexandria University.
helkamchouchi@ieee.org

² Elec. Eng. Dept, Fac. of Eng., Kafr El-Sheikh University.
eman.abouelkhair@eng.kfs.edu.eg

³ Math. Eng. Dept, Fac. of Eng., Alexandria University.
yasmine.abouelseoud@gmail.com

ABSTRACT

Electronic commerce, as exemplified by the popularity of the Internet, is becoming more important along with fast progress in communications and information technology. The SET (Secure Electronic Transaction) protocol is a scheme designed to ensure that merchants and cardholders can conduct business over a public network. Although the SET protocol has some disadvantages, but still it is the most commonly used protocol in the Internet shopping. Signcryption is a cryptographic primitive which simultaneously provides both confidentiality and authenticity in a single logical step. Signcryption based on elliptic curves provides the same level of security using smaller keys compared to schemes based on the discrete logarithm problem over finite fields. This paper examines the benefits of using signcryption rather than signature-then-encryption in the SET protocol. Using identity-based signcryption in the SET protocol reduces the number of encryption and decryption operations. Moreover, signcryption is less time consuming than signature-then-encryption.

KEYWORDS

SET Protocol, Signcryption, Elliptic Curve Cryptography

1. INTRODUCTION

Mobile payment is the transaction of fiscal values by means of mobile phones or other handheld devices. According to one of the Gartner's report [1] the total mobile users in the world will reach 7.4 billion by 2015. With such a large number of people using mobile devices, it would be increasingly used not only for communication but also as a means of monetary transactions [2]. As mobile phones have become more and more powerful with multiple features, people would rather like to have their monetary transaction done with a mobile device rather than carrying currencies and notes in their pocket. Therefore the mobile security is considered to be a major issue for mobile payment that can be faced through sensitive payment. Actually, there are many research papers discussing businesses markets, payment processing and payment schemes [3, 4], but in fact there are a few papers that deal with the construction of wireless payment schemes, involving protocols and security protection solutions [5, 6, 7].

Also, there are many existing mobile payment protocols, one of the most widely accepted mobile payment protocols the Secure Electronic Transaction protocol. Secure Electronic Transaction (SET) is a standard protocol for securing credit card transactions over insecure networks, specifically, the Internet. SET is a set of rules and regulations that enable users to perform

financial transactions through existing payment system over insecure wireless network (internet) in much secure and reliable manner [8]. SET is an application to provide various security services as confidentiality, data integrity and authenticity for all electronic transactions over the internet.

The SET protocol makes use of cryptography in order to provide confidentiality and security, to ensure payment integrity, and to authenticate both the merchant and the cardholder. Security means that merchants are protected from purchases with an authorized payment card and denial of purchases. As for cardholders, security means that they are protected from merchant imposters or theft of their payment card numbers [9]. The SET protocol, developed by Visa and Master Card, is an open standard for encryption and security specification for credit card transactions over the Internet [10].

To guarantee unforgeability, integrity and confidentiality of communications, the traditional method is to digitally sign a message with the private key of the sender then encrypt the message and the signature with a randomly chosen key using a symmetric cipher. The random key is then encrypted using the public key of the receiver. The encrypted (message + signature) is then sent together with the encrypted symmetric key. The reverse process is run at the receiver. This scheme is known as signature-then-encryption. An alternative scheme called signcryption was proposed by Zheng to simultaneously sign and encrypt messages in a single logical step with a computational cost significantly lower than that required by the traditional signature-then-encryption approach [11].

This paper discusses the SET protocol and introduces the use of signcryption in this protocol seeking performance improvements. The rest of the paper is organized as follow. An overview of the basic SET protocol and its merits is introduced in Section 2. Then, a related scheme that improves the security of the basic SET is reviewed in Section 3. In Section 4, the proposed improvement to the SET protocol based signcryption is introduced. Finally, Section 5 concludes the paper.

2. OVERVIEW OF THE SET PROTOCOL

Secure payment systems are critical to the success of E-commerce. There are four essential security requirements for safe electronic payments (Authentication, Encryption, Integrity and Non-repudiation). Security protocols adopted in electronic payment systems, such as the SSL (Secure Socket Layer) and the SET protocols, have encryption mechanisms as a key component in them [12].

2.1. The SET Protocol Purpose and Entities

2.1.1. The Purpose

The purpose of the SET protocol is to establish payment transactions that

- provide confidentiality of information;
- ensure the integrity of payment instructions for goods and services; that is, order data;
- authenticate both the cardholder and the merchant.

2.1.2. Main Entities

There are four main entities in the SET protocol; namely,

- The customer (cardholder)
- The merchant (web server)

- The payment gateway (merchant's Bank, acquirer). The payment gateway is a device operated by an acquirer. Sometimes, these two entities are treated as separate entities.
- Issuer (cardholder's bank).

Before participating in the transaction, both the customer and the merchant must obtain a digital certificate for their public keys from a certifying authority (CA).

2.2. How it Works

Both customers and merchants must register with the CA first, before they can buy or sell over the Internet. Once registration is done, a customer and a merchant can participate in electronic transactions

The protocol involves nine basic steps are described in [12,13], which are summarized below. The flow of information is depicted in Figure 1.

1. The customer browses the website of the merchant and chooses the product.
2. The merchant returns a form containing the list of items along with the total price and the order number. A copy of the digital certificate is also sent for authentication of the merchant.
3. The customer sends its signature of the order information and the payment information along with its digital certificate to the merchant. The digital certificate is to validate the customer's authenticity. The order information confirms that the customer will make the purchase, whereas the payment information is encrypted by the public key of the payment gateway which cannot be read by the merchant.
4. The merchant forwards the payment information to the merchant bank.
5. The merchant bank then forwards the information to the customer bank for authorization and payment.
6. The customer bank confirms authorization to the merchant bank and the merchant bank sends the authorization confirmation to the merchant.
7. The merchant completes the order and sends it to the customer.
8. The merchant captures the transaction from its bank.
9. The customer bank sends a notification to the customer that the payment has been processed.

Figure 1 The basic steps in the SET protocol

2.3. Merits of the SET Protocol

The SET protocol has the following merits [10,14]:

1. It provides the merchant with a protective means against online fraud.
2. As for the consumer, it guarantees the authenticity of the merchant so that credit card number will not be stolen. SET keeps more secrets for the consumer to improve the satisfaction of their on-line shopping experience.
3. It helps the bank and the credit card company to expand their service to a more broad space through the Internet and it lowers the probability of credit card on-line fraud.
4. It has defined an interface for all quarters of an online transaction so that a system can be built on the products made by the different manufacturers.

3. RELATED WORK

An improvement to the basic SET protocol has been developed by Zhang Boping, and Shang Shiyu [10] to achieve more secure electronic transactions. In the new payment model, an electronic transaction centre is introduced which has the following responsibilities: It acts as the region certifying authority, and it plays the role of the payment gateway, it ensures transaction data preservation, time-stamps transactions, and finally resolves disputes; that is, it acts as an arbitration centre.

The steps in the SET protocol have been modified as follows:

1. Customer C sends out procurement request to Merchant M.
2. Merchant M provides the inventory detail list to customer C.
3. Customer C agrees the quoted price, and then makes a purchase order. After the order is processed and digitally signed by the customer, the digital certificate is encrypted together with the digital signature of the order using the public key of the merchant M, then $CM1$ is sent to the merchant: $CM1 = E_{pkM}(D_{skC}(\text{Order}), Cert_C)$
4. The merchant M decrypts $CM1$ to get $D_{skM}(CM1) = D_{skC}(\text{Order}), Cert_C$ and thus recovers the public key certificate $Cert_C$ of customer C which is then verified. Then, M decrypts the Order using the public key of C: $\text{Order} = E_{pkM}(D_{skC}(\text{Order}))$ to confirm whether Order is sent by customer C. Merchant M encrypts its digital certificate $Cert_M$, and the payment gateway certificate $Cert_P$, as well as the payment requirement Pay : $MC2 = D_{skM}(Cert_M, Cert_P, Pay)$. The payment requirement Pay includes the name and quantity of goods, the monetary amount, the identification of customer C and merchant M as well as the time, place, and method of delivery.
5. Customer C decrypts $MC2$: $(Cert_M, Cert_P, Pay) = E_{pkM}(MC2)$ to confirm that it is sent by Merchant M. Also, the identities of Merchant M and payment gateway P are confirmed and the information including the name and quantities of goods, and the amount payable is checked. Customer C produces the payment order PI according to Pay and digitally signs it: $PI' = D_{skC}(PI)$. Then, customer C produces symmetrical key K stochastically, which is used to encrypt PI' : $PI'' = E_K(PI')$. The account information of customer C (PANIC name, credit card number and so on) and K are encrypted with the public key of the payment gateway P: $PI = E_{pkP}(PAN, K)$. Customer C carries on the signature on PI'' , PI and $MC2$: $CCA3 = D_{skC}(PI'', PI, MC2, Cert_C, Cert_M)$ then transmits all these information to the transaction authentication center CA.

6. Transaction authentication center CA decrypts $CCA3$: $(PI'', PI, MC2, Cert_c, Cert_M) = E_{pkC}(CCA3)$. $MC2$ is decrypted by the public key of merchant M : $(Cert_M, Cert_p, Pay) = E_{pkM}(MC2)$ and the $Cert_M$, $Cert_C$, $Cert_P$ are recorded into a database. After the transaction authentication center CA generates a digital signature on $Cert_M$, $Cert_C$, $Cert_P$, PI'' and PI , $CAP4 = D_{skCA}(Cert_M, Cert_C, Cert_P, PI'', PI)$, it then transmits them to the payment gateway P .
7. The payment gateway decrypts $CAP4$:

$$(Cert_M, Cert_C, Cert_P, PI'', PI) = E_{pkCA}(CAP4)$$

and confirms whether it is sent by the transaction authentication center CA . PI is then decrypted by the payment gateway $(PAN, K) = D_{skP}(PI)$ to confirm whether it is sent by customer C . K is obtained and used to decrypt PI'' : $PI' = D_K(PI'')$. The public key of customer C is used to decrypt PI : $PI = E_{pkC}(PI')$. The payment gateway obtains PI and sends PI to the credit card companies through the safe financial network.

8. Credit card companies check customer C in their databases, confirm the account has enough money, and then deposit this deal into a blocked account. After that, the credit card company will inform the payment gateway and the money will have been already deducted through the financial network.
9. Payment gateway P issues the delivery notice to merchant M : $Msg' = D_{skP}(Msg)$.
10. Merchant M decrypts Msg to confirm that it is sent by payment gateway: $Msg = D_{skP}(Msg')$. Then, merchant M delivers goods to customer C .

5. THE PROPOSED SET PROTOCOL

In this section, the proposed is described in detail. It is an enhancement to the above described protocol [10]. Full specifications of the encryption and digital signing schemes are provided.

5.1. The Set-up Phase

The customer C , merchant M , payment gateway P submit their IDs to the certificate authority that generates the secret keys and computes the public keys for itself and the other three entities: customer, merchant and payment gateway. It then sends the secret keys through a secure channel and publishes the public keys.

- CA selects two large primes p and q where $q|p-1$. An elliptic curve E is chosen with P is a generator point on the elliptic curve and a hash function $H: \{0,1\}^k \times Z_q \rightarrow Z_q$ is also needed.
- CA selects a secret number $d_{CA} \in Z_q$ and computes its public key $d_{CA} \cdot P = Q_{CA}$.
- CA computes the customer secret key: $d_C = ID_C \cdot d_{CA} \mod q$ and the customer public key: $d_C \cdot P = Q_C$.
- CA computes the merchant secret key: $d_m = ID_m \cdot d_{CA} \mod q$ and the merchant public key: $d_m \cdot P = Q_m$.
- CA computes the payment gateway secret key: $d_P = ID_P \cdot d_{CA} \mod q$ and the payment gateway public key: $d_P \cdot P = Q_P$.

5.2. The Proposed SET Protocol

The SET protocol improvement is shown in Figure 2. The SET protocol steps will be detailed in what follows.

1. Customer C sends out procurement request to Merchant M.
2. Merchant M provides the inventory detail list to customer C.
3. Customer C agrees the quoted price, and then makes purchase order. The customer signcrypts the order using his/her secret key d_C and the merchant M public key Q_m , then $CM1$ is sent to the merchant M: $CM1 = \text{Signcrypt}(\text{Order})$. The customer signcrypts the order as follows:

- generates random number w_1
- computes $w_1.Q_m = (x_1, y_1)$
- splits x_1 into k_1, k_2 .
- $r_1 = h(\text{Order}, k_2)$
- $c_1 = E_{k_2}(\text{Order})$
- $s_1 = (w_1 - r_1.d_C) \bmod q$
- sends (r_1, c_1, s_1) to the merchant M

Figure 2. The improved SET payment model

4. The merchant M unsigncrypts $CM1$ using the customer public key Q_C and his/her secret key d_m to confirm that the message is sent by a legal customer. To obtain the order, $\text{Order} = \text{Unsigncrypt}(CM1)$. The merchant unsigncrypts $CM1$ as follows:

- computes $s_1.Q_m + r_1.d_m.Q_C = (x_1, y_1)$
- splits x_1 to k_1, k_2
- $\text{order} = D_{k_1}(c_1)$
- $r_1 = h(k_2, \text{order})$

- If $r_1 = r_1^-$ the merchant accepts the signature and signs the payment requirement using his/her secret key d_m and sends it to the customer $MC2 = \text{Sign}(\text{Pay})$.

The correctness of the equation: $s_1.Q_m + r_1.d_m.Q_C = (w_1 - r_1.d_c).Q_m + r_1.d_m.Q_C$
 $= w_1.Q_m - r_1.d_c.Q_m + r_1.d_m.Q_C = w_1.Q_m = (x_1, y_1)$

The merchant performs the signing process is as follows:

- generates random number g_1
- computes $R_1 = (g_1 + d_m).P = (u_1, v_1)$
- computes $S_1 = (u_1 + \text{Pay}.d_m) \bmod q$
- sends $MC2 = (u_1, S_1, \text{Pay})$ to the customer C

Payment requirement *Pay* includes the name and quantity of goods, the monetary amount, the identification of customer C and merchant M as well as the time, place, and the method of delivery.

5. Customer verifies the signature on the *Pay* using the merchant public key Q_m : $\text{Pay} = \text{Verify}(MC2)$. The customer accepts the signature if $\text{Pay}.Q_m = (S_1 - u_1).P$. The correctness of this verification equation $\text{Pay}.Q_m = (S_1 - u_1).P$ is proven as follows: $(S_1 - u_1).P = (u_1 + \text{Pay}.d_m - u_1).P = \text{Pay}.Q_m$. Then, the customer produces the payment information *PI* that contains the account information of consumer C (PANIC name, credit card number and so on). The customer signcrypts *PI* using the public key of the payment gateway Q_p : $PI'' = \text{Signcrypt}(PI)$. The customer signcrypts the payment information *PI* as follows:

- generates random number w_2
- computes $w_2.Q_p = (x_2, y_2)$
- splits x_2 into k_3, k_4 .
- $r_2 = h(PI, k_3)$
- $c_2 = E_{k_4}(PI)$
- $s_2 = (w_2 - r_2.d_c) \bmod q$
- sends $PI'' = (r_2, c_2, s_2)$ to the merchant M

The customer signs PI'' and $CM2$ using his secret key d_C and sends it to the certificate authority CA: $CCA3 = \text{Sign}(PI'', MC2)$ where $MC2 = \text{Sign}(\text{Pay}) = (u_1, S_1, \text{Pay})$. The customer proceeds as indicated below

- generates a random number g_2
- computes $R_2 = (g_2 + d_C).P = (u_2, v_2)$
- computes $S_2 = (u_2 + (PI''//MC2).d_C) \bmod q$
- sends $CCA3 = (u_2, S_2, (PI'', MC2))$ to the certificate authority CA.

6. The certificate authority CA verifies that $CCA3$ is sent by customer C, then makes sure that $MC2$ is sent by merchant M. The procedure carried out by the CA is given below.

- The certificate authority CA performs the verification process to authenticate the customer: It accepts the signature if $(PI'' // MC2).Q_C \stackrel{?}{=} (S_2 - u_2).P$.
- The CA authenticates merchant M using merchant public key Q_m . The certificate authority CA accepts the signature if $Pay.Q_m \stackrel{?}{=} (S_1 - u_1).P$

The CA then signs PI'' using his/her secret key d_{CA} and sends it to payment gateway: $CAP4 = Sign(PI'')$. The signature generation steps are as follows:

- The CA generates random number g_3
 - computes $R_3 = (g_3 + d_{CA}).P = (u_3, v_3)$
 - computes $S_3 = (u_3 + PI''.d_{CA}) \bmod q$
 - sends $CAP4 = (u_3, S_3, PI'')$ to the payment gateway
7. The payment gateway verifies the signature $CAP4$ using the CA public key Q_{CA} and confirms whether it is sent by transaction authentication center (CA). PI'' is then unencrypted by the payment gateway using the payment gateway secret key d_P and the consumer public key Q_C . The payment gateway accepts the signature if $PI''.Q_{CA} = (S_3 - u_3).P$. The payment gateway then unencrypts PI'' , where $PI'' = (r_2, c_2, s_2)$, as follows:

- Recovers the key from: $s_2.Q_P + r_2.d_P.Q_C = (x_2, y_2)$
- splits x_2 to k_3, k_4
- $PI = D_{k_3}(c_2)$
- $r_2 = h(k_4, PI)$
- If $r_2 = r_2$, then the payment gateway accepts the signature

The correctness of the equation $s_2.Q_P + r_2.d_P.Q_C = (x_2, y_2)$ is demonstrated below.

$$\begin{aligned} s_2.Q_P + r_2.d_P.Q_C &= (w_2 - r_2.d_C).Q_P + r_2.d_P.Q_C = w_2.Q_P - r_2.d_C.Q_P + r_2.d_P.Q_C \\ &= w_2.Q_P = (x_2, y_2) \end{aligned}$$

The payment gateway obtains PI and sends PI to the credit card companies through the safe financial network.

8. Credit card companies check customer C in their databases, confirm the account has enough money, and then deposit this deal into a blocked account. After that, the credit card company will inform the payment gateway and the money will have been already deducted through the financial network.
9. The payment gateway signs the delivery notice using its secret key d_P and sends it to merchant M: Message' = Sign(Message). The payment gateway generates random number g_4 and computes:
- $R_4 = (g_4 + d_P).P = (u_4, v_4)$
 - $S_4 = (u_4 + Message.d_P) \bmod q$

Then, it sends $Message' = (u_4, S_4, Message)$ to the merchant M.

10. Merchant M verifies the received signature using the payment gateway public key to confirm that it is sent by payment gateway. It accepts the signature if $Message \cdot Q_P = (S_4 - u_4) \cdot P$.

5.3. Security Analysis

The security of the proposed SET protocol is based on the elliptic curve discrete logarithm problem (ECDLP) [15]. Up till now, the ECDLP is considered to be hard.

Definition 1: The Elliptic Curve Discrete Logarithm Problem (ECDLP) is defined as follows. Let G and Q be two points on an elliptic curve and G is of order n and n is a prime. The point $Q = k \cdot G$, where $k < n$. Given these two points G and Q, find the discrete logarithm of Q to the base G; that is, k.

The proposed scheme possesses the following properties:

1. **Unforgeability** It is computationally infeasible for an adaptive attacker to masquerade as the signcrypter in creating a signcrypted text. Signcrypted text is generated using the sender's secret key. Thus, no one can generate a valid signcrypted text without knowing the sender's secret key. Also, the ciphertext and the signature are tightly coupled in the verification phase thus if an attacker attempts to forge a signcrypted text, the verification at the receiver will fail. Therefore, forgery attack is likely not to occur. For example in step 5 if an attacker wants to generate a signcrypted text, he does the following :

- * Generate random number w'
- * $K' = w' \cdot Q_m = (x', y')$
- * Split x' into k'_3, k'_4
- * $c' = E_{k'_3}(PI')$
- * $r' = \text{hash}(m', k'_3)$
- * $s' = (w' - r' \cdot d_a) \bmod q$, d_a is the attacker secret key.
- * The attacker sends (r', c', s') to the payment gateway.

The payment gateway unsigncrypts the message by recovering the key k' as follows:

$$\begin{aligned} & s' \cdot Q_P + r' \cdot d_P \cdot Q_C \\ &= s' \cdot Q_P + r' \cdot d_P \cdot Q_C \\ &= w' \cdot Q_P - r' \cdot d_a \cdot Q_m + r' \cdot d_P \cdot Q_C \\ &\neq w_2 \cdot Q_P = (x_2, y_2) \end{aligned}$$

Without knowing the customer secret key, no attacker can generate a valid signcrypted text. Therefore, the proposed protocol achieves unforgeability.

2. **Confidentiality** It is computationally infeasible for an adaptive attacker to find out any secret information from a signcrypted text. The message is encrypted with a symmetric key cipher that is generated by the sender using the receiver's public key and only a valid receiver can recover the key using his secret key. Therefore, finding out any secret information from a signcrypted text is computationally infeasible. For example

when the customer signcrypts the order information in step 3 and sends (r_1, c_1, s_1) to the merchant M. Only, a valid merchant with d_m can recover the key $k = s_1 \cdot Q_m + r_1 \cdot d_m \cdot Q_C = (x_1, y_1)$. If the attacker wants to derive the original message, he must be able to recover the randomly generated session key k . The secret key used to encrypt the message, k_1 , is part of the x-coordinate value of the point K . However, the extraction of the secret key k_1 is equivalent to solving the ECDLP. Assume that the attacker tries to compute point $k = s_1 \cdot Q_m + r_1 \cdot d_m \cdot Q_C = (x_1, y_1)$, he should derive the receiver's secret key d_m , where $Q_m = d_m \cdot P$, therefore to derive d_m one needs to solve the ECDLP. Without knowing the secret key of the receiver, no attacker can recover the message encryption key. It is only the valid receiver with secret key d_m who can recover the key and unsigncrypt the message.

3. **Non-repudiation:** It is computationally feasible for a judge to settle a dispute between the signcrypter and the recipient in an event where the signcrypter denies the fact that he is the sender of the signcrypted text to the recipient. If the sender denies sending the message, the receiver sends the recovered key K to a trusted third party who decrypts the message then hashes it with the key and checks if the claimed sender is the origin of the message or not.

5.4. Comparative Study

When comparing the proposed protocol with the protocol in [10], it is apparent that there is no need for the certificates in the proposed protocol because the customer, merchant and payment gateway send their IDs to the CA that generates the key pair for each entity. This way the key management procedure is simplified and the need for verifying CA signatures over digital certificates is eliminated. This reduces bandwidth requirements as well as the computational burden associated with the implementation of the SET protocol.

Moreover, the scheme in [10] uses signature-then-encryption to provide authentication and confidentiality. On the other hand, the proposed scheme uses signcryption and this further reduces the computational time. Table 1 shows the time abbreviations that will be used in the comparison table.

Table 2 shows the comparison between the protocol in [10]; that is based on signature-then-encryption, and the proposed protocol; that is based on the signcryption. The comparison shows that the proposed protocol is more computationally efficient than that in [10]; where the former uses four signature operations, five signature verifications, two signcryption operations and two unsigncryption operations but the latter uses six signature operations, seven signature verifications, four encryption operations and four decryption operations. This means that the proposed protocol saves the time required for two encryption and decryption operations. Moreover, signcryption is less time consuming than signature-then-encryption.

Table 1. Time abbreviations

Symbol	Operation
$T_{EC-mult}$	time required for executing a multiplication operation on an elliptic curve E
T_{EC-add}	time required for executing an addition operation on an elliptic curve E
T_{mult}	time required for executing a modular multiplication in a finite field
T_h	time required for executing one way hash function operation
T_{enc}	time required by the system for executing an encryption operation
T_{dec}	time required by the system for executing a decryption operation
T_{sig}	time required by the system for executing a signing operation
T_{verify}	time required by the system for executing the signature verification
T_{si_cry}	time required by the system for executing a signcryption operation
T_{unsi_cry}	time required by the system for executing an unsigncryption operation

Table 2. The comparison of the proposed protocol with the protocol in [10]

Step	Protocol in [10]	The Proposed protocol
3	$1T_{sign}+1T_{enc}$	$1T_{si_cry}$
4	$1T_{dec}+1T_{sign}+1T_{verify}$	$1T_{unsi_cry}+1T_{sig}$
5	$1T_{verify}+2T_{sign}+2T_{enc}$	$1T_{verify}+1T_{si_cry}+1T_{sign}$
6	$2T_{verify}+1T_{sign}$	$2T_{verify}+1T_{sign}$
7	$2T_{verify}+2T_{dec}$	$1T_{unsi_cry}+1T_{verify}$
9	$1T_{sign}$	$1T_{sign}$
10	$1T_{verify}$	$1T_{verify}$
Total	$6T_{sig}+7T_{verify}+4T_{enc}+4T_{dec}$	$4T_{sig}+5T_{verify}+2T_{si_cry}+2T_{unsi_cry}$

Moreover, when the proposed protocol is compared with the protocol in [16], it is found that the protocol in [16] uses the basic SET protocol discussed in Section 2, but using signature-then-multiple-encryption by the DES scheme. On the other hand, the proposed protocol uses the signcryption primitive that is fast and more secure than the protocol in [16].

5.5. Performance of the Proposed Protocol

The performance analysis of the proposed protocol is presented in Table 3, it provides the count of the computationally expensive operations involved in the protocol as a whole. In the proposed protocol, the customer performs 5 scalar point multiplications over an elliptic curve, 2 encryptions, 3 hash operations and 3 multiplications over a finite field. As for the merchant, it performs 5 scalar point multiplications and one addition operation over an elliptic curve. It also requires one hashing operation and 2 multiplications over a finite field. The payment gateway performs 5 scalar point multiplications and one addition operation over an elliptic curve. Additionally, it carries out one decryption operation, one hashing operation and 2 multiplications over a finite field. Finally, the certifying authority performs 5 scalar point multiplications over an elliptic curve, 2 encryption operations and one multiplication over a finite field.

Table 3. The performance of the proposed protocol

Step	The proposed
3	$1T_{EC-mult}+1T_{enc}+1T_h+1T_{mult}$
4	$4T_{EC-mult}+1T_{dec}+1T_h+2T_{mult}+1T_{EC-add}$
5	$3T_{EC-mult}+1T_{enc}+1T_h+2T_{mult}$
6	$5T_{EC-mult}+1T_{mult}$
7	$4T_{EC-mult}+1T_{dec}+1T_h+1T_{mult}+1T_{EC-add}$
9	$1T_{EC-mult}+1T_{mult}$
10	$2T_{EC-mult}$
Total	$20T_{EC-mult}+2T_{enc}+2T_{dec}+4T_h+8T_{mult}+2T_{EC-add}$

6- CONCLUSION

This paper introduced an efficient secure electronic transaction (SET) protocol based on signcryption, which is more efficient than signature-then-encryption. The use of an identity-based infrastructure eliminates the need for digital certificates. This simplifies the key management process and reduces both the required bandwidth and the computational time associated with the SET protocol. The proposed protocol achieves various security requirements through the use of an efficient signcryption scheme. The signature part is unforgeable, and the message is only readable to its intended recipient achieving confidentiality.

The proposed protocol outperforms other schemes in literature and thus promotes its use in practical scenarios. It is superior to the protocol in [10] as it reduces the number of encryption and decryption operations required. Moreover, resorting to signcryption provides further savings in the computational time required by the proposed protocol. Moreover, the proposed protocol is also more efficient than the protocol in [16] that employs signature-then-multiple symmetric encryption using DES.

REFERENCES

- [1] C. PETTEY, "GARTNER SAYS WORLDWIDE MOBILE CONNECTIONS WILL REACH 5.6 BILLION IN 2011 AS MOBILE DATA SERVICES REVENUE TOTALS \$314.7 BILLION", 2011 [ONLINE] AVAILABLE AT [HTTP://WWW.GARTNER.COM/IT/PAGE.JSP](http://www.gartner.com/it/page.jsp) (ACCESSED 21 FEBRUARY 2012).
- [2] M. S. DING AND C. R. UNNITHAN, "MOBILE PAYMENTS (MPAYMENTS) –AN EXPLORATORY STUDY OF EMERGING ISSUES AND FUTURE TRENDS", INFORMATION TECHNOLOGY AND ORGANIZATIONS DEAKIN UNIVERSITY, 2002 .[ONLINE] AVAILAIBLE AT [HTTP://WWW.IDEA-GROUP.COM](http://www.idea-group.com), (ACESSED 15 FEBRUARY 2012),pp. 99-101.
- [3] L. ANTOVSKI, AND M. GUSEV, "M PAYMENT ", PROCEEDINGS OF THE 25TH INTERNATIONAL CONFERENCE INFORMATION TECHNOLOGY INTERFACES, 2003 (ITI'03).
- [4] S. NAMBIAR, AND T.L. CHANG, "M PAYMENT SOLUTIONS AND M COMMERCE FRAUD MANAGEMENT", RETRIEVED SEPTEMBER 9, 2004 FROM [HTTP://EUROPA.NVC.CS.VT.EDU/~CTLU/PUBLICATION/M PAYMENT SOLUTIONS.PDF](http://europa.nvc.cs.vt.edu/~CTLU/PUBLICATION/M%20PAYMENT%20SOLUTIONS.PDF)
- [5] X. ZHENG, AND D. CHEN, "STUDY OF MOBILE PAYMENTS SYSTEM", PROCEEDINGS OF THE IEEE INTERNATIONAL CONFERENCE ON E COMMERCE , 2003 (CEC'03).
- [6] S. KUNGPISAN, B. SRIVNIVASAN, AND P.D. LE, "A SECURE ACCOUNT BASED MOBILE PAYMENT PROTOCOL", PROCEEDINGS OF THE INTERNATIONAL CONFERENCE ON INFORMATION TECHNOLOGY: CODING AND COMPUTING, 2004 (ITCC'04).

- [7] A. FOURATI, H.K.B. AYED, F. KAMOUN, AND A. BENZEKRI, "A SET BASED APPROACH TO SECURE THE PAYMENT IN MOBILE COMMERCE", IN PROCEEDINGS OF 27TH ANNUAL IEEE CONFERENCE ON LOCAL COMPUTER NETWORKS (LCN'02), NOVEMBER 06 08, 2002, TAMPA, FLORIDA
- [8] WIKIPEDIA: THE FREE ENCYCLOPEDIA, TECHNICAL WEBLINK: [HTTP://EN.WIKIPEDIA.ORG/WIKI/SECURE_ELECTRONIC_TRANSACTION#HISTORY_AND_DEVELOPMENT](http://en.wikipedia.org/wiki/Secure_Electronic_Transaction#History_and_development)
- [9] [HTTP://MRAJACSE.FILES.WORDPRESS.COM/2012/04/SECURE-ELECTRONIC-TRANSACTION-SET.PDF](http://mrajacse.files.wordpress.com/2012/04/secure-electronic-transaction-set.pdf)
- [10] Z. BOPING , AND S. SHIYU," AN IMPROVED SET PROTOCOL", ISBN 978-952-5726-02-2 (PRINT), 978-952-5726-03-9 (CD-ROM), PROCEEDINGS OF THE 2009 INTERNATIONAL SYMPOSIUM ON INFORMATION PROCESSING (ISIP'09) , HUANGSHAN, P. R. CHINA, AUGUST 21-23, 2009, PP. 267-272
- [11] Y. ZHENG, "DIGITAL SIGNCRYPTION OR HOW TO ACHIEVE COST (SIGNATURE AND ENCRYPTION) COST (SIGNATURE) + COST (ENCRYPTION)", ADVANCES IN CRYPTOLOGY, LNCS, VOL. 1294. SPRINGER-VERLAG, PP.165–179, 1997.
- [12] Y. LI AND Y. WANG, "SECURE ELECTRONIC TRANSACTION (SET PROTOCOL)", [ONLINE] AVAILABLE AT [HTTP://WWW.PEOPLE.DSV.SU.SE/~MATEI/COURSES/IK2001_SJE/LI-WANG_SET.PDF](http://www.people.dsv.su.se/~matei/courses/IK2001_SJE/LI-WANG_SET.PDF) (LAST ACCESSED 15 FEBRUARY 2012).
- [13] R. BORGOHAIN, M. T. SINGH, C. SAKHARWADE AND S. SANYAL," TSET: TOKEN BASED SECURE ELECTRONIC TRANSACTION"; INTERNATIONAL JOURNAL OF COMPUTER APPLICATIONS, [ACCEPTED] MAY,2012, ISBN: 978-93-80866-55-8, PUBLISHED BY FOUNDATION OF COMPUTER SCIENCE, NEW YORK, USA, DOI: 10.5120/5056-7374.
- [14] S. LU, S. SMOLKA, "MODEL CHECKING THE SECURE ELECTRONIC TRANSACTION (SET) PROTOCOL," PROCEEDINGS OF THE 7TH INTERNATIONAL SYMPOSIUM ON MODELING, ANALYSIS AND SIMULATION OF COMPUTER AND TELECOMMUNICATION SYSTEMS, 1999:358-364.
- [15] D. JOHNSON, A. MENEZES, AND S. VANSTONE, " THE ELLIPTIC CURVE DIGITAL SIGNATURE ALGORITHM (ECDSA) ",INTERNATIONAL JOURNAL OF INFORMATION SECURITY 1 (1) (2001) 36–63.
- [16] H. GUPTA AND V. K. SHARMA," ROLE OF MULTIPLE ENCRYPTION IN SECURE ELECTRONIC TRANSACTION", INTERNATIONAL JOURNAL OF NETWORK SECURITY & ITS APPLICATIONS (IJNSA), VOL.3, NO.6, NOVEMBER 2011.