SET: A QUESTIONABLE SECURITY PROTOCOL

Charles A. Shoniregun and Songhe Zhao
School of Computing & Technology, University of East London
4-6 University Way Docklands, London
United Kingdom, E6 5NJ

Keywords: Secure Electronic Transaction (SET), Dual Signature, Elliptic Curve Cryptography (ECC).

Abstract: The Secure Electronic Transaction (SET) was developed by Visa and MasterCard in 1997. The SET is a protocol that is theoretically perfect with very high expectation to provide secured electronic financial transactions. It also provides a 'dual signature' as it hides credit card numbers from the merchants, and purchase details from the bank. This paper exploits the weaknesses that led to SET's failure and proposed SET's encryption process with elliptic curve cryptography (ECC).

1 INTRODUCTION

The SET is a standard protocol for ensuring the security of electronic financial transactions on the Internet, which has been endorsed by virtually all the major players in the electronic commerce arena, including IBM, MasterCard International, Visa International, Microsoft, Netscape, GTE, ViriSign, SAIC and Terisa. The SET defines a detailed secure transaction process among all participants. It mainly replies on the sciences of cryptography to implement its functions for securing electronic transactions on the Internet. The cryptography provides two different encryption mechanisms and an authentication mechanism. SET employs DES symmetric encryption and RSA asymmetric encryption to provide functions of data encryption, digital signature and digital envelope, which can provide guarantee for the security of information transmitted over the Internet. To enable merchants to verify transactions, SET put the public key into an electronic document (i.e. digital certificate). The certificate is then signed by a trusted third-party, such as certificate authority (CA), which can then be verified from their certificate and so on in a hierarchy of trust. And it will also protect buyers by providing a mechanism for their credit card number to be transferred directly to the credit card issuer for verification and billing without the merchant being able to see the number. With SET, a user is given an electronic wallet (digital certificate) and a transaction is conducted and verified using a combination of digital certificates and digital signatures among the purchaser, a merchant and the purchaser's bank in a way that ensures authenticity and privacy.

2 RELATED WORK

The SET handles processes for once-off merchant and card holder registration, for making a purchase and getting the payment authorized through a bank's network gateway and for subsequent payment to the merchants. SET pilots have been conducted around the world, but very few have yet been implemented. The SET protocol provides four main advantages for securing data/information transfer:
Confidentiality: using two forms of cryptography---DES and RSA to protect transmitted messages from intercepting.
Integrity: using one-way cryptographic hashing algorithms and digital signatures to make sure that the messages transmitted have not been modified in transit.

Authentication: using X.509v3 digital certificates which assure that the parties involved in the transaction are who they claim to be, and prevent them from denying that they sent a message (i.e. non-repudiation).

- Privacy: using cryptography to make sure the information is only available to parties in a transaction when and where necessary.

A. Shoniregun C. and Zhao S. (2007).
SET: A QUESTIONABLE SECURITY PROTOCOL.
In *Proceedings of the Third International Conference on Web Information Systems and Technologies - Internet Technology*, pages 313-319
DOI: 10.5220/0001291903130319

There have been remarkable efforts aimed at the SET's weaknesses. These efforts are effective in overcoming the weaknesses and reducing redundancies. Abbott (1999) presented that although SET offers a complete card payment system that manages financial risk and defines interoperability, it requires that software be installed in the banking network, at merchants' locations, and on consumers' PCs. This deployment obstacle has slowed and complicated the adoption. Weishaupl et al. (2006) developed gSET as a solution for the unsolved problems in the field of dynamic trust management and secure accounting in commercial virtual organizations. The gSET establishes trust and privacy between entities in a Grid environment by adapting the concept of SET used for electronic credit card transfers in e-Commerce. However, this solution is known as an enabling step to make Grids a platform for commercial workflows but it is not a solution to address problems of SET's adoption and deployment. Bella et al. (2005) produced an accurate formal model to identify the SET protocol goals and then to prove them. In their study, they were troubled by the complexity of SET protocol so that they are not clear whether model checking could cope with this SET's complexity. it is very clear that the complexity of SET is the crucial problem for its adoption. This rather serious time lag issue could come to be resolved by another more efficient method called elliptic curve security. Although RSA employed by SET is the most popular public-key cryptosystem today, the long-term trends such as the proliferation of smaller, simpler devices and increasing security needs will make continued reliance on RSA more challenging over time.

3 DIGITAL SIGNATURE

The SET uses digital signatures and certificates stored in an electronic "wallet" on the users' personal computers and on the merchants' websites. The dual signature which is employed as the SET's digital signature is issued by a trusted third party certificate authority and the user's certificate contains card details. It is encrypted in such a way that it can only be read by the card issuer, not the merchant.

The dual signature is generated by creating the message digest of both messages sent by a sender, concatenating the two digests together, computing the message digest of the result and encrypting this digest with the sender's private signature key. A recipient of either message can check its authenticity by generating the message digest on its copy of the message, concatenating it with the message digest of the other message (as provided by the sender) and computing the message digest of the result. If the newly generated digest matches the decrypted dual signature, the recipient can trust the authenticity of the message (Secure Electronic Transaction LLC, 1997). Privacy and Authentication are achieved through the use of digital signatures. Digital signatures are aimed at achieving the same level of trust as a written signature has in real life. This helps achieve non-repudiation, as the consumer cannot later establish that the message wasn't sent using his private key (IBM Corporation, 1998).

3.1 Construction of Dual Signature

In SET, when a customer has place an order from a merchant's website and is going to make a payment for the order, he/she needs to send the order information (OI) to the merchant and payment information (PI) to the merchant's acquirer through the merchant. In the progress of sending OI and PI to destinations, the customer needs to use hash algorithm (SHA-1) to produce message digest (MD) respectively for OI and PI. These two MDs are then concatenated and then computed to payment and order message digest (POMD) by using the hash algorithm again. Finally, the customer encrypts the POMD with the customer's private signature key (KRc) in order to create the dual signature (DS). The operation formula is:

$$DS = E_{KR_c}\left[H(H(PI)\,||\,H(OI))\right]$$

When the merchant is in the possession of OI, PIMD and DS, and PI, the merchant can use the customer's public key (KUc), which is taken from the customer's certificate to compute two quantities. The operation formula is:

$$H(PIMD\,||\,H(OI))$$

and

$$D_{KU_c}[DS]$$

If these two quantities are equal, the merchant has verified the signature. Similarly, when the acquirer is in the possession of PI, OIMD and DS, the acquirer can use KUc to compute other two quantities. The operation formula is:

$$H(H(PI)||OIMD)$$

and

$$D_{KU_c}[DS]$$

Again, if these two quantities are equal, the acquirer has verified the signature. In summary:

- The merchant has received OI and verified the signature.
- The acquirer has received PI and verified the signature.
- The customer has linked the OI and PI and can prove the linkage.

Within SET, dual signatures are used to link an order message sent to the merchant with the payment instructions containing account information sent to the Acquirer. When the merchant sends an authorization request to the Acquirer, it includes the payment instructions sent to it by the cardholder and the message digest of the order information. The Acquirer uses the message digest from the merchant and computes the message digest of the payment instructions to check the dual signature.

4 CRICISM OF SET AND ECC-BASED IMPROVEMENT

Although the advantages of SET had ever attracted our sight, this focus quickly moved to its weaknesses and the relevant researches were carried out soon after the failure of SET happened.

4.1 Weaknesses of SET

The SET protocol has the potential of securing the electronic financial transactions over the Internet so as to enable e-Commerce to be safe and attractive to consumers, but unresolved problems and issues which caused the failure of employing SET as a dominate protocol in e-Commerce as follows:

i *Complex and slow processing:*
The SET models all of the players involved in electronic financial transactions. It is a complex protocol because it so totally simulates these existing real world processes. The ultimate high level of security that SET provides is implemented by a very complex system which causes the processing transactions to be quite slow. Generally, the access operations to the merchant's server can approach 6 times (Zhao, 2005); moreover, SET also implements a deal of computing for public key encryption. The workload of the merchant's server hereby is quite heavy so that overload is most likely to happen. Lag times of up to 50 seconds have been reported for the processing of a typical cardholder initiated purchase request to the approval response from the acquirer and the finalization of the transaction by the merchant's server (Keenan, Disenso and Green, 1998). This matter is critical for average SET's consumers because the key advantage of e-Commerce is to provide ease-of-use applications for its participants. Nobody would like to wait for a response around 50 seconds after sending a purchase request. A common download time for a Webpage should be around 15 seconds and if it above 30 seconds then the customer will stop waiting and move over to another Website (Shoniregun, 2005), not even for a very attractive Webpage. As a result, it's not recommended that that any response over 15 seconds is proper, especially since there is little or no feedback to reassure and update the cardholder about the progress of the transaction processing (Wolrath, 1998).

ii *Inconvenient and expensive deployment and implementation:*
To implement SET, merchants are required to invest in new software and build their businesses around a complex transaction infrastructure, but it doesn't make them jump with joy (Friedman, 1998). Merchants who want to benefit from SET have to install SET-enabled applications in their systems. It doesn't look inconvenient and expensive to the installation in a single merchant's system. However, e-Commerce is a global electronic transaction over the Internet. It may create considerable profit only if a great number of people step into this virtual business world to play this game. Therefore, it's necessary for absolute majority of merchants to install SET-enabled applications in their systems. This needs a huge investment and long time to deploy far and wide, but most people don't want to be at risk before making an investment and seeing other's implementation. The latter slows down the deployment of SET's, not only to merchants, but also to the installation of SET-enabled applications in client end is also obstacle for SET's implementation. Ease-of-use

is the key to e-Commerce as well as SET. But the serious question remains about just how intuitive it will be for users to install the necessary applications to do SET-based transactions. Neither Microsoft nor Netscape are in any rush to build SET into their browsers until there is a demand for it. But merchants won't support it until their customers ask for it, and their customers won't ask for it until it's built into their browsers. Although Vendors like VeriFone and IBM have developed SET wallet plug-ins for the popular Web browsers, Internet users are not keen on using external plug-ins. Because they have to install SET from the external plug-ins that adds payment functionality to a browser, and also register with a financial institution or trusted third party, which then issues digital certificates that identify the cardholder to the merchant and vice versa. Although in the long run SET backers expect certificates to be included in wallets -- which in turn will be built into browsers.

Generally speaking, different versions of SET software are currently available in the market but it is critical that these packages interoperate. Without interoperability, any protocol is dead. SET defines interoperability between all parts of the card-payment process. A system can be built with parts from multiple vendors. To offer these benefits, SET requires that software be installed in the banking network, in merchants' systems, and on consumers' PCs. This "deployment obstacle" has slowed the adoption of SET, which also requires that certificates be issued to all parties. These requirements make SET quite inconvenient and expensive to deploy all over the world.

iii *Lack key and certificate management:*
Today's popular operating systems are unreliable and insecure, so they are highly vulnerable to attack, particularly when connected to the Internet. For this reason, non-repudiation really exists only when private signing keys are kept out of untrusty PCs (Abbott, 1999). But SET refers nothing about how to securely keep keys and certificates out of attacks after finishing an electronic transaction. It appears that these will need to be stored on participants' workstations and servers, or additional peripherals installed on workstations and servers to handle a secure token (Clark, 1996). A secure token, such as a smart card, is a good tool to store and exercise private signing keys and certificates for its holder. It can also provide an easy way for signing keys and certificates to be issued and carried around. The SET was designed specifically for smart cards, and all SET wallets support tokens. This beneficially ties token directly to e-Commerce, because the certificate authenticates the customer as a cardholder and directly signifies that transaction is taking place. However, smart cards are distributed very slowly, because no one has solved the problem of how to get card readers attached to everyone's PC. But other form-factor tokens are appearing, including small universal-serialbus tokens (key fobs, for example) that can do the same job. Some SET pilots also are being conducted without tokens, with the expense of the added complexity of distributing and managing certificates as software files. In fact, the smart card's set-up is beyond the average cardholder. Actually it's a very troublesome procedure.

4.2 ECC-based Improvement of SET

Our proposition is based on substituting RSA algorithm by using ECC to provide SET's cryptography (See Figure 1). Our laboratory experiments have proved that ECC provides greater security and more efficient performance than the first generation public key techniques.

i *ECC from the consumer:*
The consumer use the ECC hash algorithm (SHA-1) to produce message digest for the original message and encrypt the message digest with ECC private key to produce the digital signature. This generates a random symmetric key that encrypt the original message, digital signature and a copy of the consumer's certificate, which contains public signature key. The symmetric key is encrypted by using the merchant's ECC public key-exchange key and the encrypted key (digital envelope) will be sent to the merchant along with the encrypted message. Sending a set of message to the merchant consists of the symmetric encrypted original message, as well as the asymmetrically encrypted symmetric key (the digital envelope).

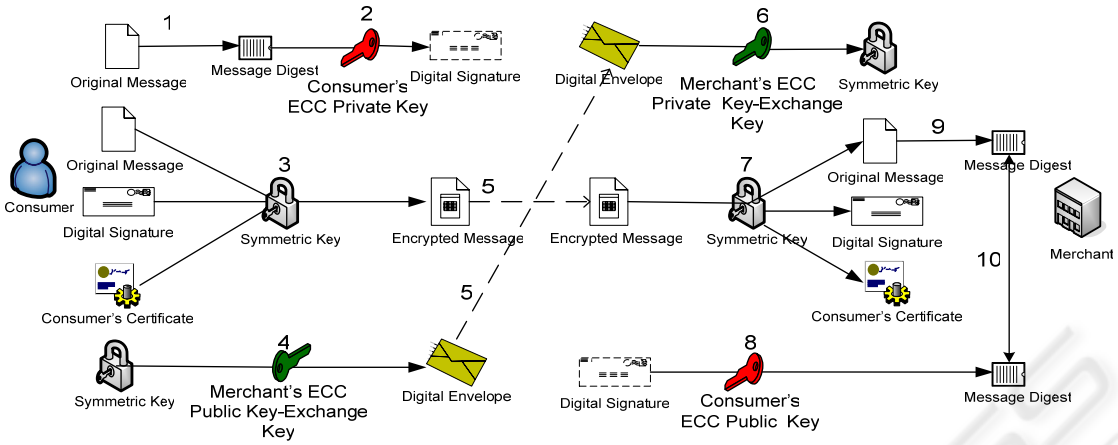

Figure 1: SET's encryption process with ECC.

ii *ECC from the merchant:*
Receiving the set of message from the consumer and decrypting the digital envelope with the ECC private key-exchange key to retrieve the symmetric key. The consumer's digital signature and certificate would be required in other to decrypt the original message. Decrypting the consumer's digital signature with the ECC public key would enabled the original message digest to produce a new message digest of the decrypted original message and compare the message digest with the one obtained from the consumer's digital signature.

The ECC enhance the process in the SET's digital signature and inverse the process to provide the message digest by decrypting the digital signature. Moreover, the process related to the digital envelope is also served by this solution. As a result of this ECC-based solution, the duration of SET's encryption process can be reduced much more than RSA-enabled process. The discussion section focuses on how the security level of the encryption will advance according to the features of ECC.

5 DISSCUSSION

In comparison with RSA, ECC offers the same level of security using much smaller keys but faster computations and less resourceful on memory.

Table 1: Comparison of key sizes.

Symmetric Key Size (bits)	RSA and Diffie-Hellman Key Size (bits)	Elliptic Curve Key Size (bits)
80	1024	160
112	2048	224
128	3072	256
192	7680	384
256	15360	521

The Table1 shows the key sizes that protected the keys used in conventional encryption algorithms like the (DES) and (AES) together with the key sizes for RSA, Diffie-Hellman and elliptic curves that are needed to provide equivalent security.

Notably, the use of 1024-bit RSA does not match the 128-bit or even 112-bit security level now used for symmetric ciphers in SET. This indicates the need to migrate to larger RSA key sizes in order to deliver the full security of symmetric algorithms with more than 80-bit keys. However, RSA Laboratories stated that 1024-bit RSA to be unsafe for data that must be protected beyond 2010 and recommends larger keys for longer term protection (Kaliski, 2003). In the case of providing higher key sizes, RSA will increase much more the time cost of the encryption process in SET. For example, employing RSA or Diffie-Hellman to protect 128-bit AES keys should use 3072-bit parameters, which is three times longer than the size of 1024-bit in use throughout the Internet today. However, the

equivalent key size for elliptic curves is only 256 bits. It's obvious that as symmetric key sizes increase the required key sizes for RSA and Diffie-Hellman increase at a much faster rate than the required key sizes for ECC. Hence, ECC offer more security per bit increase in key size than either RSA or Diffie-Hellman public key systems.

Table 2: Relative computation costs of Diffie-Hellman and elliptic curves.

Security Level (bits)	Ratio of DH Cost: EC Cost
80	3:1
112	6:1
128	10:1
192	32:1
256	64:1

Besides providing better security, ECC also are more computationally efficient than the first generation public key systems, RSA and Diffie-Hellman. Although elliptic curve arithmetic is slightly more complex per bit than either RSA or DH arithmetic, the added strength per bit more than makes up for any extra compute time (See Table 2). The following table shows the ratio of DH computation versus EC computation for each of the key sizes listed in Table 1.

Closely related to the key size of different public key systems is the channel overhead required to perform key exchanges and digital signatures on a communications link. The key sizes for public key in Table 1 (above) is also roughly the number of bits that need to be transmitted each way over a communications channel for a key exchange. In channel-constrained environments, elliptic curves offer a much better solution than first generation public key systems like Diffie-Hellman.

In choosing an elliptic curve as the foundation of a public key system there are a variety of different choices. The National Institute of Standards and Technology (NIST) has standardized on a list of 15 elliptic curves of varying sizes. Ten of these curves are for what are known as binary fields and 5 are for prime fields. Those curves listed provide cryptography equivalent to symmetric encryption algorithms with keys of length 80, 112, 128, 192, and 256 bits and beyond. In our solution, it recommends that SET employs the ECC with 256, 384, and 521 bits. It is expected to show enhanced speed on the SET payment gateway and potential cost savings. It also promises to lower the capability threshold for small devices to perform strong cryptography, and increase a server's capacity to handle secure connections (Lenstra and Verheul, 2001).

6 CONCLUSION

The SET security guarantee is achieved by using the cryptography but is a very costly technology. The three main weaknesses of SET have been discussed and we have also proposed the ECC model to enhance not only the higher level security but also to perform more efficient than RSA. Our future work will focus on the SET key and certificate management.

REFERENCE

Abbott, S., (1999). 'The debate for secure E-commerce', *UNIX Review's Performance Computing, Vol 17, Iss 2, pp. 37*

Bella, G., Massacci, F. and Paulson, L.C., (2005). 'An overview of the verification of SET', *International Journal of Information Security*, Heidelberg (1615-5270), Vol.4, Iss.1-2, pp.17

Clark, R., (1996). '*The SET Approach to Net-based Payments'* [online], Available from: http://www.anu.edu.au/people/Roger.Clarke/EC/SETOview.html [Accessed 16 September 2006]

Kaliski. B., (2003). 'TWIRL and RSA Key Size' [online], *RSA Laboratories Technical Note*. Available from: http://www.rsasecurity.com/rsalabs/technotes/twirl.html [Accessed 25 September 2006]

Keenan, V., Disenso and Green, (1998). '*PROMISES: What ever happened to SET?'* [online], Available from: http://www.herring.com/mag/issue51/promises.html [Accessed 28 September 2006]

Friedman, M., (1998). 'SET standard not exactly hitting the fast lane', *Computing Canada*, Vol 24, Iss 23, pp 26

IBM Corporation, (1998). '*An overview of the IBM SET and the IBM CommercePoint Products'* [online], Available from: http://www.software.ibm.com/commerce/set/Over--view.html [Accessed 11th September 2006]

Lenstra, A. and Verheul, E., (2001). 'Selecting Cryptographic Key Sizes', *Journal of Cryptology,* Vol. 14, pp. 255-293.

National Security Agency, (unknown). '*The Case for Elliptic Curve Cryptography'* [online], Available from: http://www.nsa.gov/ia/industry/crypto_elliptic_curve.cfm [Accessed 03 October 2006]

Shoniregun, C.A., (2005). '*Impacts and Risk Assessment of Technology for Internet Security: Enabled Information Small-Medium Enterprises (TEISMES)',* USA: Springer, pp. 14-30

Weishaupl, T., Witzany, C. and Schikuta, E., (2006). 'gSET: Trust Management and Secure Accounting for Business in the Grid', *Proceedings of the Sixth IEEE International Symposium on Cluster Computing and the Grid (CCGRID'06) - Volume 00,* pp. 349-356

Secure Electronic Transaction LLC, (1997). '*SET Secure Electronic Transaction Specification: Book 1 Business Description – Version 1',* pp. 12-29

Stallings, W., (2002). *Introduction to Secure Electronic Transaction*, USA: Prentice Hall.

Wolrath., E., (1998). '*Secure Electronic Transaction: a market survey and a test implementation of SET technology',* Master thesis, Uppsala University.

Zhao, Q., (2005). '*Network Security and Electronic Commerce',* China: Tsinghua University Publications, pp. 171-225