マルチメディア通信と分散処理　90－11
コンピュータセキュリティ　2－11
（1998．9．18）

インターネット上のモバイルコンピュータのための エージェントに基づく安全な支払いシステム

易 訓, 岡本 栄司

北陸先端科学技術大学院大学情報科学研究科

概　要

SET 及び、モバイルエージェントの規則に基づく SET/A は、モバイルコンピュータ環境を用いたインターネット上の支払い要求に適応するよう開発された。潜在的な脅威が存在する環境からエージェントを守るため、SET/A はエージェントが起動している商業サーバの安全な実行環境に依存している。しかしながら、この方法で安全性を確立するのは困難である。本論文では、SET/A と同様にインターネット上におけるモバイルコンピュータのためのエージェントに基づく支払いシステム（我々はこれを SET/A+と呼ぶことにする）を提案する。このシステムは、認証局にわずかなタスクを付加することによってエージェントの実行環境による安全性の制限を取り除くことができる。SET/A+は SET と同レベルの安全性を持ち、SET を使ったオンライン支払いシステムの代替方法を提供することができる。

A Secure Agent-Based Payment System for Mobile Computing on Internet

Xun Yi, Eiji Okamoto

School of Information Science

Japan Advanced Institute of Science and Technology

Abstract

SET/A, guided by the SET rules and based on the mobile agent paradigm, has been lately developed to meet the requirements of Internet payment in mobile computing environments. In order to protect an agent from potential malicious environment, SET/A depends on a secure execution environment in merchant's server for an agent to run. However, the security is hard to achieve. In this paper, we propose another secure agent-based payment system for mobile computing on Internet (namely SET/A+), which removes the limitation for the security of agent execution environment only by attaching an extra slight task to the certificate authority. SET/A+ is able to ensure the same level security as SET, providing an alternative way for on-line payment using SET protocol.

1 Introduction

The recent burgeoning of new communications technologies and, in particular, the Internet explosion have brought electronic commerce to the brink of widespread deployment. However, businesses are wary about treading beyond that brink, largely because of concerns about unknown risks they may face. The key to alleviating many of these concerns – to mitigating the risks – is security.

Almost every Internet user has heard of credit card frauds, performed by hackers eavesdropping connections used to send those data - despite the fact that very few of those attacks have actually succeeded. Even the deployment of secure servers, based on protocols such as SSL or S-HTTP, is not enough, since the credit card information is deposited in the server, where it can easily be read by anyone with access to it (even if not authorized).

The concern for protecting the user's credit card information lead VISA and MasterCard, in association with major software and cryptography companies, into the development of the SET protocol [1]. SET protocol, in particular its purchasing phase, is intended for users connected to the Internet during an entire transaction. This requirement can not be easily satisfied in mobile computing environments. SET/A [2], lately presented, guided by the SET rules and based on the mobile agent paradigm, is able to take away the computational burden from the user's device, so it can be disconnected while the transaction is running. However, SET/A depends on a secure execution environment in merchant's server to protect an agent's confidential data (i.e., credit card information) against malicious merchant. In fact, the security is hard to ensure.

In this paper, we propose another secure agent-based payment system for mobile computing on Internet (namely SET/A+), which takes away the limitation for the security of agent execution environment in merchant's server only by attaching an extra slight task to the certificate authority. SET/A+ is able to ensure the same level security as SET, providing an alternative way for on-line payment using SET protocol.

2 Description of SET/A+

SET/A+ is implemented with an agent travelling from the cardholder's computer, carrying all the protected relevant information, to the merchant's server. On arrival, the agent performs the cardholder's role and carries on a complete purchase request transaction with the merchant.

A purchase request under SET/A+ has a few more steps than in SET, since an agent has to be sent to the merchant's server, and come back to the cardholder's computer when the transaction is done. The process of SET/A+ purchase request can be illustrated in the following figure:

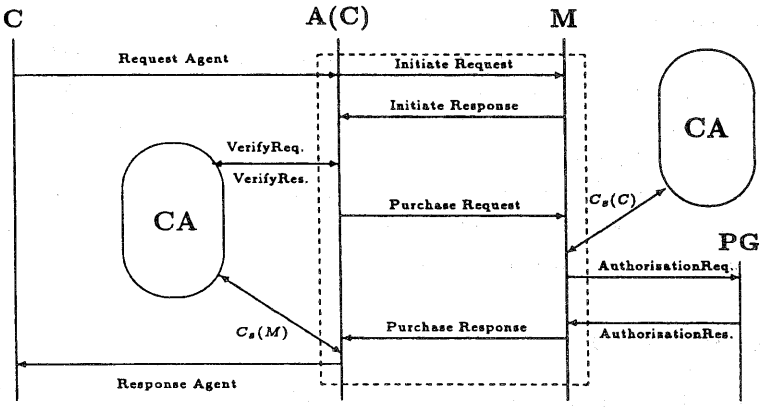

Fig.1. SET/A+ purchase request transaction

Note 1 The above process involves three participants: Cardholder (C), Merchant (M) and Payment Gateway (PG). A(C) represents the agent which resides in the merchant server and talks with M on behalf of C.

Note 2 As SET, each participant in above process possesses two kinds of certificates, one for key-exchange (whose public key is contained in certificate C_k, the pair of key-exchange public-secret key of entity A are denoted as (A_p, A_s)), which is used for encrypting and decrypting operations, another for signature (in certificate C_s), used for creation and verification of digital signatures. SET/A+ employs Digital Signature Standard to create and verify signature. The pair of signature public-secret key of entity A are denoted as (y_A, x_A), where $y_A = g^{x_A} (mod\ p)$, p is a large prime, g is an integer which has order $q\ (mod\ p)$.

Note 3 SET/A+ specifies a hash function (H) to generate a digest of message.

Note 4 SET/A+ is designed to be as compatible with SET as possible, only requiring slight significant modifications. The merchant software could remain unchanged, since its interaction with the agent is mostly the same as it would be with the cardholder. The only exception is that it must be aware that now it's talking to an entity residing in a host other than the cardholder.

Under the above assumptions, the detailed procedure in Fig.1 can be described as follows:

Step 1. An user, namely the cardholder (C), looks at a catalog (printed in paper, supplied in a CD-ROM or available on-line on the Web) of a company, namely the merchant (M) and, after deciding to purchase something, builds a request with the same elements as in the original SET request. Then, a request agent, A(C), is sent to the merchant's server, carrying the following contents:

- Request, the description of the services or the quantities of the goods, the terms of the order, and the brand of the credit card that will be used for payment.
- $C_s(C)$, the cardholder's signature certificate.
- The Order Information OI, containing control information verified by the merchant to validate the order, card brand and bank identification. The OI also includes an order description, which includes the amount of the transaction and other elements such as quantity, size and price of the items ordered, shipping and billing addresses, etc.
- The Payment Instructions PI, containing the amount of the transaction, the card account number and expiration date, instructions for installment payments (if that's the case). The PI is encrypted with a randomly generated symmetric key K_C, i.e., $E_{K_C}(PI)$, where the bit length of K_C should be shorter than that of the prime p.

  Both OI and PI will contain the unique transaction identifier I_C assigned by C.
- The digest of PI (i.e., H(PI)) and the dual signature for the OI and PI, i.e., cardholder's signature for $H(H(OI)\|H(PI))$, denoted as $Sign_C[H(H(OI)\|H(PI))]$.
- The data for the certificate authority CA,

$$z = y_{CA}^{t \cdot x_C} (K_C + I_C \cdot t + r)\ (mod\ p) \tag{1}$$

where r is an integer number randomly chosen from $[1, p-1]$ and t is the current time. In a word, the agent involves:

$$Request, C_s(C), I_C, OI, E_{K_C}(PI), H(PI), Sign_C[H((H(OI)\|H(PI))], z, r, t \quad (2)$$

Step 2. After arriving at the merchant's server, the merchant verifies the agent and then supplies an agent execution environment for the agent to run. The agent resides in the environment and sends (now locally) an initiate request (i.e., Request in the agent) to the merchant M.

Step 3. On basis of the initiate request, especially the brand of the credit card, the merchant returns an initiate response, i.e., a signed message with its signature certificate $C_s(M)$, the payment gateway key-exchange certificate $C_k(PG)$, and the merchant unique transaction identifier I_M to the agent. The payment gateway is a device operated by financial institute with which the merchant established an account for processing payment with the brand used by the cardholder.

Step 4. The agent contacts the certification authority (CA) by transmitting a verification request, which is composed of

$$C_s(M), C_k(PG), C_s(C), z, t, I_M \quad (3)$$

CA verifies $C_s(M)$ and $C_k(PG)$ and then computes:

$$\begin{aligned} & (y_C^{t \cdot x_{CA}})^{-1} \cdot z \\ = \; & (g^{x_C \cdot t \cdot x_{CA}})^{-1} \cdot g^{x_{CA} \cdot t \cdot x_C} \cdot (K_C + I_C \cdot t + r) \\ = \; & K_C + I_C \cdot t + r \ (mod\ p) \end{aligned} \quad (4)$$

and returns the agent with a confirmation which includes:

$$PG_p(K_C + I_C \cdot t + r + I_M \cdot t) \quad (5)$$

After receiving the above confirmation, the agent creates the digital envelope EN_{PG} for the payment gateway, which contains the following two items:

$$PG_p(K_C + I_C \cdot t + r + I_M \cdot t), I_C, I_M, t, r, E_{K_C}(PI) \quad (6)$$

The purchase request, including

$$EN_{PG}, C_s(C), OI, H(PI), Sign_C[H(H(OI)\|H(PI))] \quad (7)$$

is then sent to the merchant.

Step 5. The merchant verifies the certificate and the dual signature on the OI with OI, H(PI) and $Sign_C[H(H(OI)\|H(PI))]$. The request is then processed, which includes forwarding the digital envelope, H(OI) and $Sign_C[H(H(OI)\|H(PI))]$ to the payment gateway, for authorization. If I_C, I_M, r and t are compatible, the payment gateway should be able to obtain K_C and then PI from EN_{PG}. If OI and PI are agreeable, the dual signature on PI can be verified with H(OI), PI and $Sign_C[H(H(OI)\|H(PI))]$. After processing the order, the merchant generates and signs a purchase response, and send it to the agent along with its signature certificate. If the payment was authorized, the merchant will fulfill the order, by delivering the products bought by the cardholder.

Step 6. The agent receives the response, verifies the certificate and the signature, and migrates back to the cardholder's computer.

Step 7. The agent arrives at the cardholder's host, carrying the response from the merchant. The cardholder's software then proceeds as in SET's final step.

In step 6 we haven't detailed how the rendezvous takes place. One possible approach is to use a mechanism similar the one used by cellular phone operators to deliver SMS messages when the user re-connects.

3 Security Issues

In a network payment system, one of the obvious concerns is to protect the user's critical data, in particular the credit card information. SET's usage of the dual signature mechanism and the encryption of the PI and account information (into a digital envelope with the payment gateway's public key-exchange key), ensure the necessary privacy of the critical data. In particular, the data is protected from a potentially hostile environment, such as the merchant server. Of course, protecting the merchant from a malicious agent is also important, but that concern is clearly outside the scope of this paper.

If we relax the requirement of following closely the SET protocol for purchase requests, there may be a better way to achieve the goal of protecting the data. First, recall that the data we want to protect from the merchant is to be encrypted with the payment gateway's key. If the cardholder knows in advance which payment gateway the merchant is using for the card brand, and if the OI and the PI can be built in advance, then the process of generating the dual signature, the random key, and the digital envelope can be performed before the agent leaves the cardholder's computer. When it migrates, all the information, completely protected, can now go with it, and the agent only has to give it to the merchant and wait for the response. This approach has one major drawback: to obtain the payment gateway's certificate in advance, the cardholder has to perform an initial request to the merchant, wait for the response, and then proceed with the agent.

For SET/A+ to be able to ensure the same level of protection as SET, without modifying SET too much, it must be possible for the agent to carry classified information without having to disclose it to the wrong entities. Also, the generation of the symmetric key K has to be performed in such a way that no one other than the cardholder and the payment gateway has knowledge of it.

Based on the above considerations, SET/A+ requires the cardholder randomly chooses the symmetric key K_C and encrypts the payment information with the key himself before the agent leaves the cardholder. Although the certificate authority is required firstly to retrieve $(K_C + I_C \cdot t + r)$ as (4) and then re-encryptes it with the public key of the payment gateway as (5), it has no knowledge of K_C in view of the existence of random number r. Furthermore, CA has no access to the order information and payment information.

The merchant software of SET/A+ still remains unchanged. Though the merchant can scan the agent to hold all information in the agent, he can not retrieve $(K_C + I_C \cdot t + r)$ from (1) or (5) because it has been encrypted firstly with the public key of the certificate authority and

then with the public key of the payment gateway. Therefore, he has no knowledge of the K_C and the cardholder's payment information.

Slightly different from SET, the symmetric key K_C can not directly obtained by decryption in the authorization step of SET/A+. However, only a few simple algebraic operations are needed to compute out K_C, but it can bring an unexpected advantage to authorization, i.e., the cardholder's transaction identifier, the merchant's transaction identifier, and the transaction time must be compatible, otherwise, the resulted symmetric key is not correct and the payment information can not be retrieved. Therefore, SET/A+ can efficiently prevent reply attacks (making the agent pay more than one) from the hostile merchant.

4 Conclusion

SET is expected to gain wide acceptance as a secure Internet payment system since it combines the well-known credit card payment method with an elaborated security protocol. However, SET is a very complex and "heavy" protocol and it is not practical to use in mobile computing environments because the low bandwidth and high connection costs are generally associated with mobile computing. In view of it, SET/A, based on the SET protocol and the mobile agent model, has been lately proposed. In order to protect an agent from potential malicious environment, SET/A depends on a secure execution environment in merchant's server for an agent to run. However, the security is hard to achieve.

In order to remove the limitation for the security of agent execution environment, a novel secure agent-based payment system for mobile computing on Internet (namely SET/A+) is proposed as above. Only by attaching an extra slight task to the certificate authority, SET/A+ is able to ensure the same level security as SET, providing an alternative way for on-line payment using SET protocol.

We are also interested in keeping the agent as intelligent and autonomous as possible, allowing it to take its own decisions (even if very simple) when needed. As part of our future work, we intend to use a mobile agent system to implement SET/A+, with agents capable of negotiating with their hosts, based on the knowledge they carry as they migrate from one server to another.

Acknowledge

I would like to appreciate Mr.Takeshi Okamoto for his help with translating the abstract from English to Japanese.

Reference

1. Visa International and MasterCard International, "Secure Electronic Transaction (SET) Specification", Version 1.0, May 1997.

2. Artur Romao and Miguel Mira da Silva, "An Agent-Based Secure Internet Payment System for Mobile Computing", TrEC'98, Hamburg, 3-5 June 1998, LNCS Series, Springer-Verlag.