COMPUTERS & SECURITY 28 (2009) 289–300

available at www.sciencedirect.com

ScienceDirect

journal homepage: www.elsevier.com/locate/cose

Computers & Security

Secure multiparty payment with an intermediary entity☆

Mildrey Carbonell[a,*], *José María Sierra*[a], *Javier Lopez*[b]

[a]*Universidad Carlos III de Madrid, Avenida de la Universidad, #30, 28911, Leganés, Madrid, España, Spain*
[b]*University of Malaga, Avda. Cervantes, 2 29071, Málaga, España, Spain*

ARTICLE INFO

Article history:
Received 25 April 2007
Received in revised form
11 December 2008
Accepted 11 December 2008

Keywords:
Electronic payment
Secure multiparty protocol
Intermediation
Electronic commerce

ABSTRACT

During the last years, many secure electronic payment solutions have been proposed but most of them are focused on the traditional two-party business models with a customer and just one provider. In this paper we propose a new secure multiparty payment model with an intermediary, who helps the customer to make purchases and payments with many providers simultaneously. In our secure infrastructure it is assumed that the intermediary does not need to be a trusted entity (it does not need to be a TTP). One of the most important issues of this contribution is the Intermediary-3D: we propose a simple adaptation of the 3D Secure™ payment protocol in order to maintain the 3D Secure™ working modes but offering the possibility of making multiple secure payments through an intermediary. By means of this slight adaptation, our model avoids the provider's enrolment process in a centralized system (e.g. Visa Domain) and it makes more robust and secure the multipayment scenarios, as well as, it favors its deployment in global networks like Internet.

1. Introduction

Electronic commerce is now one of the widest applications in Internet since it helps businesses to expand their marketing strategy and to reduce their costs. This growth has motivated the development of research to improve electronic services. Security, as one of these research topics, constitutes a critical point in the implementation of new business models because the process of traditional business such as paper-based contracts, personal purchases, etc. must be adapted to flows of information inside an unreliable network like the Internet. Payment should be the process with the highest security level in e-commerce operations because it is the step where the customer legally ends the business by making the money transference.

Many secure electronic payment solutions have been proposed. Some of them describe online payment with a cash payment model, like e-Cash, DigiCash, NetCash, and Cybercash. Others, such as NetBill, NetCheque and BankNet, present a cheque payment model. And, in a card payment schema, open solutions such as iKP and SET have been developed as a standard of secure payment. iKP and SET were not widely used in the Internet but they constitute a starting point in the development of secure payment solutions. Today, the most popular solution in the card payment schema is the 3-D Secure™ protocol (3-D Secure) developed by VISA and MasterCard, which is based on the ideas of iKP and SET. This protocol provides the card issuer with the ability of authenticating its cardholders during an online purchase. Given that VISA has licensed this protocol and that many vendor

☆ This work was partially supported by *i-aspects* Project (Reference Models for Secure Architectures in Mobile Electronic Payments), TIN2007-66107.

* *Corresponding author.*
E-mail addresses: mcarbone@inf.uc3m.es (M. Carbonell), sierra@inf.uc3m.es (J.M. Sierra), jlm@lcc.uma.es (J. Lopez).

0167-4048/$ – see front matter © 2008 Elsevier Ltd. All rights reserved.
doi:10.1016/j.cose.2008.12.002

communities use it, 3D Secure™ is considered a standard for authenticated payment.

Although there are many payment solutions, most of them (including 3D Secure™) are focused on the traditional two-party business models with a customer and a provider. However, many business models involve some intermediary entities to help the process. For example: auction models (www.ebay.es) need entities (Chen, 2004; Lee and Jun Lee, 2006) that handle the offers and the bids and buy and sell models (http://www.buysell.com) use entities to analyze the offers and the demands. In Internet business models, the intermediaries have been represented as middle applications for publicity services (such as virtual mall, marketplace, e-procurement and so on) (Rappa, 2004). They also have been represented as an automatic agent of searching, shopping behaviors simulator (Jenamani et al., 2003; Yarom et al., 2003), adjustments of price (Wang et al., 2004), calculating market strategies (Lau, 2006), taking decision (Sen et al., in press; Li et al., 2006) and optimizing the order (Lin and Lin, 2006) (inside the agent-based e-commerce). However, in almost the case, the intermediary has been described, in the payment process, as a secure payment gateway (www.paypal.com) or as a trusted third party (Kim and Lee, 2003; OPELIX Project; Tsai et al., 2002).

In this paper, we describe a multiparty electronic commerce protocol in which the intermediary plays the role of a payment mediator. This intermediary helps the customer to make purchases and payments with many providers simultaneously as a single payment transaction. Our proposed model decreases the number of customer operations in the traditional multiparty payment process. This optimization in the payment process for this kind of multiparty scenarios is particularly interesting when we deal with devices which have some resources constraint (computational or connectivity), this is the case of portable devices. Also, in the secure infrastructure proposed, is not assumed to have strong trusting restrictions in the intermediary entity (i.e. not need to be a TTP) which implies a more flexible scenario. In addition, we propose an adaptation of the 3D Secure™ payment protocol, using our intermediary, to offer the possibility of making secure payment with multiple providers that not need to be enrolled in VISA 3D Secure.

This paper is organized as follows. In Section 2, we present some comparative reviews of intermediary entities, their main functionalities in the payment process and security solutions; we also define our model with an intermediary entity and its security requirements. In Section 3, we present our secure protocol to handle multipayment. Next in Section 4 we describe the extension of the 3D Secure™ model. In Section 5, some advantages and real applications of our proposal are presented. Finally, in Section 6, the conclusions are presented and some future work for our proposal.

2. Intermediary entities in e-commerce

2.1. *Related work*

In the OPELIX Project the intermediary is described as an entity within the business process between customer and provider. The project presents five models (called incremental business models) where the provider gradually delegates the business phases (advertising, negotiation, ordering, payment and delivery) to the intermediary. Although security not a goal of the OPELIX project, the authors, in Hauswirth et al. (2001) describe some security mechanisms to implement each model. In the payment model, the paper proposes the use of a TTP to register all money transactions or to implement the intermediary as a trusted entity.

Other work (Wang and Li, 2004) presents the intermediary as a mobile agent between the customer and the provider. It describes a secure protocol (called **LITESET/A++**) to delegate the signature of contracts and payment processes to the agent. The protocol employs a TTP and some cryptography techniques (*signature-share scheme* and *signcryption-share scheme*) to protect the transactions and to avoid fraudulent behaviors on the part of the agent. This solution, like all the solutions based on TTP, has many security and implementation problems (election of the reliable entity, bottlenecks, network delays …). Moreover, this solution is poorly accepted by the customer because the agent has the responsibility of selecting the final products.

Others secure mechanism, as **proxy signature** (Lia et al., 2003; Wang et al., 2007; Yu et al., 2008), proposes the idea of a intermediary entity between the sender and the receiver This mechanism has been integrated in many payment scenarios, although its main disadvantage is the complexity for analyzing its security properties. In fact, security flaws have been found in some of these solutions (Gou and Wang, 2007), short time after its publication.

The intermediary in multiparty models is described in the project **COYOTE** (Tsai et al., 2002) as an improvement on the business services of Virtual mall. It describes an infrastructure by which the virtual mall allows the customer to buy on the Internet from multiple stores. This solution describes the intermediary as a secure coordinator who can authenticate the clients on all stores. This paper is limited to covering implementation aspects of this kind of service.

Some works, such as Sans and Agnew (2001), present an extension of Secure Electronic Transaction (**SET**) based on multi-payment schema (we call **MultiSET**) where it is not required the TTP participation. This proposal describes a coordination and signature of k merchants, where one random merchant (un-trusted entity) acts as intermediary entity to coordinate the multi-signature process. This paper work with (n, k) threshold signature and require that a set of k merchants perform together a cryptographic action.

An adaptation of the intermediary to a peer-to-peer network (**P2P payment**) is proposed in Onieva et al. (2003). In this paper, the intermediary is presented as a semi-trusted agent created by the merchant to collect payment details and move it through the P2P network.

In one of our previous work (Onieva et al., 2004), we presented a **non-repudiation** protocol with an intermediary. We introduced the intermediary entity to facilitate the collection, verification and storage of non-repudiation evidence on behalf of the originator. This solution, like most of the non-repudiation solutions, employs the TTP to publish the key and to create submission evidence. In this contribution we only focus on covering the non-repudiation security requirements.

A security improvement to the multiparty model (one customer and many providers) is proposed in Wang and Varadharajan (2005). This work is an extension of the **LITESET/A++** protocol where the agent has autonomy in customer operations, but now with multiple providers (we called **LITESET/A++-multi**). This proposal keeps a TTP in all the multiple transactions and a semi-trusted between customer and agent.

Finally, in the Liberty Alliance Project (2007) (project to development open standards where consumers, citizens, businesses and governments can conduct online transactions while protecting the privacy and security of identity information) the intermediary is also described as an entity inside the business structure. However, the Liberty Alliance intermediary profile is described as an entity in charge of controlling the relationships, managing security and sharing information about the performed process. This characterization has more similarities with the representation of the intermediary as a secure payment gateway or as a trusted third party.

2.2. Our model

Our entities model (Fig. 1) are the customer (*C*), the new intermediary (*IN*), multiple providers (P_i for $i = 1..NP$ *where NP is Number of providers*), the issuer (*I*), multiples acquirer banks (A_i), and the intermediary bank (B_{IN}). Furthermore, it also includes, the payment system (*PS*) (such 3D-Secure, see Fig. 2), which is the infrastructure that performs payment transactions on behalf of the involved parties (issuer *I* and the acquirer *A* on the Internet side and *C* and *P* on the private banking network side). This relation is represented by the arrow between those entities and the *PS*.

The intermediary stores a list of products for each of the providers ($Pd_{i,j}$ *for* $i = 1..NP$ and $j = 1..NPd_i$ *where* NPd_i *is the number of products for this Provider* P_i). The customer can order products (through the intermediary) from one or many providers. The intermediary acts as mediator between customer and providers, facilitating multi-purchases and multi-payment. The customer delegates the multiple transactions to the intermediary and performing a single secure transaction among them.

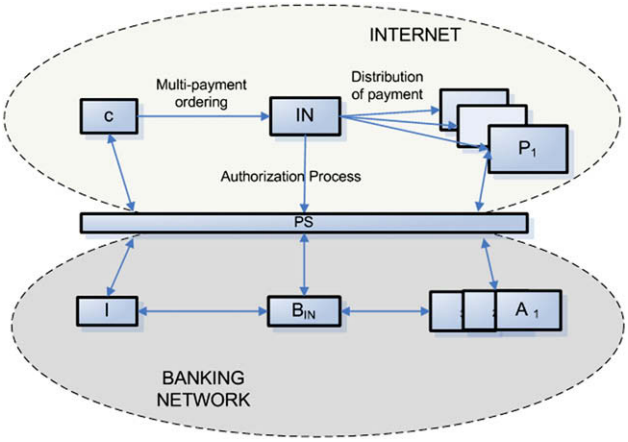

Fig. 1 – Multiparty payment with intermediary.

In that way the payment process starts with a single transaction between *C* and *IN* (represented in the figure as the arrow ***multi-payment ordering***). This one includes all the payment information to be sent to the providers such as: amount of purchases, issuer identification of the customer, payment instruments, etc. Next, the *IN* performs multiple transaction (represented in the figure as the arrow ***distribution of payment***) with the involved providers.

We consider the possibility that the intermediary implements an interface with a payment system (represented in the figure as the arrow ***authorization process***) and it could handle the payment authorization process on behalf of the customer and providers, even though the providers do not implement them.

Finally, the *IN* bank entity stands for the profits of the intermediary due to its facilitation role in the payment process. These profits could be obtained in an offline payment process for a contract with the involved entities (in the figure represented by the arrow between *IN* bank and the *A* and the *I*), or as an online payment of each transaction or offered service (represented by the arrow between B_{IN} and the *PS*). Due to these profits we believe that is not possible to consider the intermediary as a TTP in any secure solution to this model.

We think that our model fits into any Internet large and distributed business infrastructure, such as virtual mall, auctions, agent-based commerce, marketplace, etc. It could be adapted to applications in which *C* has no network connection and uses the intermediary as a network access point. This situation is typical in airports, subways, etc. where it is necessary to implement proxies for connecting with the outside network.

This situation also motivates to avoid or to move the high computational cost operations towards other entities, preserving the architecture security properties. Furthermore, this consideration is even more important in the development of the m-commerce where mobile devices (mobile phones, PDA, etc.) could have limited computational resources (Téllez and Sierra, 2007).

2.3. Security requirements

Like all electronic payment operations (Tsiakis and Sthephanides, 2005), the security requirements of our electronic payment model are *confidentiality*, *integrity*, *authentication* and *non-repudiation*.

Confidentiality protects private information from unauthorized access. *Integrity* guarantees that the information is reliable; in transaction, it guarantees that the data received is equal to the data sent. Both are reached using combinations of cryptography functions (*symmetric encryption algorithm* for confidentiality and *digest functions* for the integrity). They need to implement robust mechanisms for key distribution. With a public key infrastructure (PKI), these security requirements can be reached. In our proposal we assume that there is a PKI established in the system.

Authentication guarantees the identity of a user; in a transaction, it guarantees the identity of the issuer of the message and the integrity of the message. In electronic payment, it is important to authenticate the customers with their purchases and the providers with their products. Most of the authentication solutions use digital signature and identity certificates

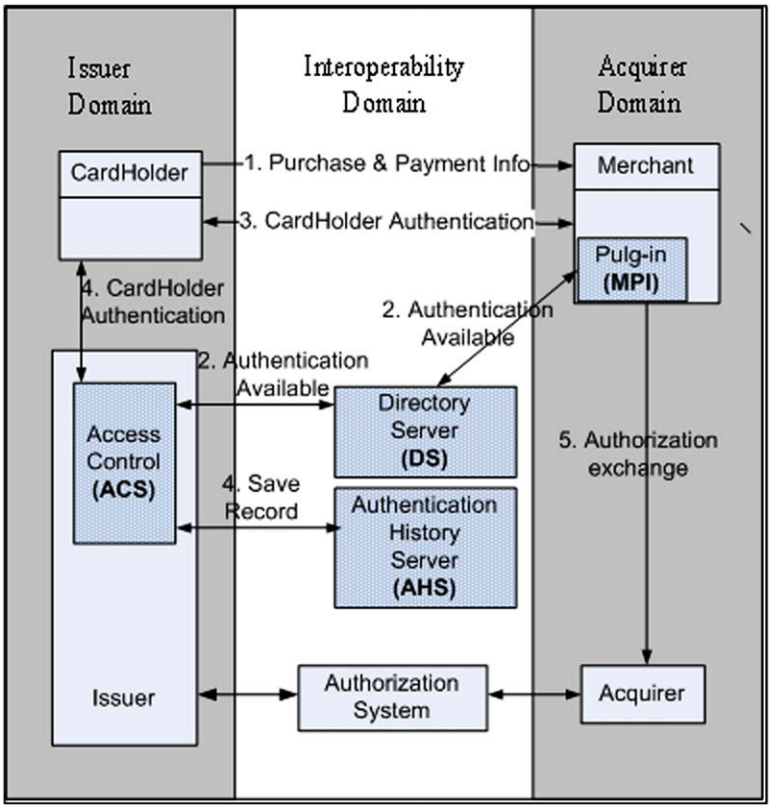

Fig. 2 – 3D Domain of VISA.

for each of the parties. In our model, the intermediary facilitates payment between multiple providers, so a new mechanism of customer authentication is required. The customer is authenticating using only one signature for all the information (purchases, payment info). The authentication mechanism must avoid malicious actions (such as identity falsification, data modifications and so on) of the intermediary in the distribution process. Also, in the model we must generate an evidence of intermediary's participation especially in applications that involved a commission.

Non-repudiation must ensure that no party can deny having participated in a part or in the whole of the protocol. So, a non-repudiation protocol must generate cryptographic evidence in support of a resolution of a dispute. In typical non-repudiation protocols, at least two types of evidence must be collected by the participating entities – non-repudiation of origin and non-repudiation of recipient. In our model the intermediary entity is involved in non-repudiation services and acts as the originator sending messages to multiple recipients, and also acts as recipient of messages. So, new types of evidence are needed – non-repudiation of intermediary origin, non-repudiation of intermediary recipient. We proposed a solution in Onieva et al. (2004).

3. Security infrastructure

An intuitive solution to our model is that the customer signs the purchases and payment info for each provider, so the intermediary acts as proxy, distributing the message among the providers. This intuitive solution has clear limitations to our model where we would like the intermediary to assist the business. In our payment solution we aim at the following goals.

- **Goal 1** (*Privileges of managing the payment*): The intermediary receives an authorization to distribute the payment. With this authorization the customer does not need to be online in the distribution process. This authorization must have a time limit.
- **Goal 2** (*Evidence of intermediary participation*): The intermediary needs evidence of his participation in the payment process, for example by means of his signature in the distribution process. This evidence is important for avoiding disputes in applications for which the intermediary receives a commission.
- **Goal 3** (*Few customer operations*): The customer creates one signature for all the providers and sends it to the intermediary. It must include the purchases and payment info and must avoid fraudulent behavior on the part of the intermediary.

Our security approach consists in creating a short-term certificate as authorization to distribute the payment information. This certificate is used for delegating privilege of payment from the customer to the intermediary and creates an ***evidence of intermediary participation*** in the payment

Notation	
$SubPd_{ij}$	Subset of purchases by C from provider P_i where $1<i<NP$ y $1<j<Npd_i$)
Kp_X	The public key of entity X
Kr_X	The private key of entity X
$E_K(M)$	Encryption of message M, with key K
$S_X(M)$	Digital signature of entity X on message M, with the key associated to the identity certificate.
$S_X(M)_Y$	Digital signature of entity X on message M with a public key Y
$H(M)$	Hash function of M

process. It also enables the provider to obtain the ***customer authentication*** and ***assurance of purchases integrity***.

The following basic notation is used throughout the section.

3.1. *Payment authorization for the intermediary*

Our short-term certificate (*ST_x509*) is similar to a public key certificate x.509 (PKC) but the valid time is very short due to security restrictions. This certificate is used to create the evidence of intermediary's participation which will be the *Subject* and it is signed by the customer (*Issuer*), who acts as the certificate authority (CA). The public key must be generated by *IN*. (in the security analysis section, we explain the reason for using this key instead of using the intermediary's PKC).

The field *ValidityPeriod* has the valid time of authorization. This time represents the interval [*not before, not after*] where the *IN* must distribute the payment. For calculating this time, it is important to consider: the length of the key (in order to avoid that the key could be cracked) and the communication delay time between involved parties (Carbonell et al., 2004).

Finally, the certificate includes some *Extensions* will be used as authorization to *IN* (*Extension*). This one has the **security info** necessary for the distributing process and will be used to validate the customer and purchases (see the Payment distribution Protocol in Section 3.2 for a more detailed explanation).

Security info

- **List of Hashes** (*ST_x509.Extension.Hashes*): It stores a list of hashes of the purchases by each provider. This means that $ST\_x509.Extension.Hashes = [P_i,\ H(SubP_{i,j})]$ where $1<i<NP$ and $1<j<Npd_i$ and $SubPd_{i,j}$ is the subset of purchases from the provider P_i. The intermediary has permission to distribute only this subset of purchases.
- **Timestamp of purchases** (*ST_X509.Extension.Tp*): It stores the Date and Time of customer's purchase.
- **Identifier of purchases** (*ST_x509.Extension.ID*): It stores the unique identifier of the purchases, generated by the customer, and it could be a result of applying a hash function to the purchases, the identifier of *IN*, and the timestamp of purchases. This means that $ST\_x509.Extension.ID = H(SubPd_{i,j}, IN, Tp)$.

Following we describe its representation inside the x509v3.

```
Short-Term certificate (x509.v3) {
   Issuer = C
   Subject = IN
   Validity
    {Notbefore, Notafter: valid time of authorization}
   SubjectPublicKeyInfo
    {…
    key: Public key generated by IN}
   Extensions
    {Hashes {…
    extnValue = [P_i, H(SubP_i,j)]: List of Hashes}
    Tp {…
    extnValue = TIME: Timestamp of purchases}
    ID {…
    extnValue = H(SubPd_i,j, IN, Tp): Identifier
    of purchases}
    }
   …}
```

3.2. *Payment distribution protocol*

Following we present the flow of messages for the payment distribution and how the intermediary uses the *ST_x509*. The protocol does not intentionally mention such purchase information as price; amount and so on, to simplify the description of the protocol. Also, we do not describe the payment authorization process because we are interested in representing a general solution which will be connected with some different payment system (such as 3D secure).

We assume that the temporal key of the intermediary (to include in the certificate) was sent, in an initial process (a time before of the current execution of the protocol using a secure exchange key protocol). The intermediary must send the key signed, to prove his identity. The security analysis of this protocol is described in Section 3.3.

Some new notations are:

- *PI*: Purchase Identifier.
- *P′*: Provider involved in the purchases and payment process
- *Td*: The timestamp of distribution process.
- $EvI_i = S_I(SubPd_{i,j}, Td)_{ST\_x509.key}$: ***Evidence of intermediary participation*** in the distribution process for each $P_i \in P'$. The intermediary signs this evidence using the public key that appears in the certificate.
- *InfoPayment*: Payment info such as cardholder, issuer, account number and so one. Since this information is sensitive; it could be signed and encrypted for the banks.

Protocol

$C \rightarrow I$: *C, I, PI,* $SubPd_{i,j}$, *InfoPayment, ST_x509 where* $1<i<NP$ *and* $1<j<Npd_i$

$I \rightarrow P_i$: *I,* P_i, *ST_x509,* $SubPd_{i,j}$, EvI_i *For each* $P_i \in P'$

The protocol works in the following way

- **Step1**: *C* sends all the purchases $SubPd_{i,j}$, the *InfoPayment* and the certificate *ST_x509*. This field *PI* is the same as that in *ST_X509.Extension.ID*

- **Step2**: The intermediary verifies the authenticity of *C* by means of *ST_X509* signature, and the purchase integrity, checking the hashes. He then distributes $SubPd_{i,j}$, *ST_X509* and his evidence of participation EvI_i for each involved P_i. *C* need not be online because *ST_X509* has all the security info for his identity and purchases. It is impossible for an intermediary to modify the *ST_X509* or create a new one because it is signed by *C*.

At the end of the protocol, the providers can check the ***customer authentication*** by validating the certificate signature (*ST_X509.SignatureValue*). The ***authenticity and integrity of the purchases*** can be checked by validating the appropriate entry *i* in the field *ST_X509.Extension.Hashes*[i].

3.3. *Protocol analysis*

In this section, we analyze our protocol using the approach of *realistic* and *accurate* analyses proposed in Bella and Bistarelli (2004).

3.3.1. *Accurate analyses: (based on the description of the goals achieved by our protocol)*

1. **Goal 1** (Privileges of managing the payment):
   The customer creates a short-term certificate authorizing the intermediary to distribute the purchases and payments. This short-term certificate includes the identifier of the intermediary and the interval of time [*not before, not after*] within which the intermediary must distribute the payment. The provider can check this authorization by means of the customer's signature in the short-term certificate.
2. **Goal 2** (Evidence of *IN*'s participation):
   The intermediary generates EvI_i as evidence of participation. This evidence is generated in the distribution process for each $P_i \in P'$.
   a) The provider P_i can check the evidence of the intermediary by checking the signature in EvI_i using the public key *ST_X509.Extensions.key*
3. **Goal 3** (Few customer operations):
   The customer only needs to sign the *ST_X509* and then send it to the intermediary. The intermediary must resend this *ST_X509* to each $P_i \in P'$. The intermediary and the providers can check the authenticity of the customer and his purchases as follows.
   a) The intermediary can obtain the identity certificate of the customer from the PKI and:
      - Check *ST_X509* signature for **customer authentication**.
      - Calculate $H_i = H(SubPd_{i,j})$ for each $P_i \in P'$.
      - Compare H_i with *ST_X509.Extensions.Hashes*[i] for each $P_i \in P'$ in order to prove the ***purchase integrity***.
   b) The provider $P_i \in P'$ can obtain the identity certificate of the customer from the PKI and:
      - Check *ST_X509* signature for ***customer authentication***.
      - Check the authorization of the intermediary *IN* inside *ST_X509.Holder*.
      - Check *IN*'s signature over EvI_i using *ST_X509.Extensions.Key* for ***evidence of intermediary participation***
      - Compare $H(SubPd_{i,j})$, where $SubPd_{i,j}$ is the received purchase, with the appropriate *ST_X509.Extensions.Hashes*[i] in order to prove the ***purchase integrity***.

3.3.2. *Realistic analysis: (based on the description of threats). Prevention of fraud by intermediary*

a) *False purchases*: If the intermediary tries to send false purchases (creating or modifying $SubPd_{i,j}$) using a legitimate customer. This means:
   $I \rightarrow P_i$: *I*, P_i, *ST_X509*, $SubPd'_{I,j}$, EvI'_I *For some* $P_i \in P'$

Solution: P_i could compare $H(SubPd'_{i,j})$, where $SubPd'_{i,j}$ is the received purchase, with the appropriate *ST_X509.Extensions.Hashes*[i]. If those values are not equal, P_i stops the protocol and declares this fraud. The security of the value inside *ST_X509* is based on the public key signature.

b) Replay of purchases: The intermediary tries to send the same purchases and with the same *ST_X509* inside the validity period (*ST_X509.attrCertValidityPeriod*). this means:
   i.$I \rightarrow P_i$: *I*, P_i, *ST_X509*, $SubPd_{i,j}$, $EvI_{i,z}$ n.$I \rightarrow P_i$: *I*, P_i, *ST_X509*, $SubPd_{i,j}$, $EvI_{i,z}$ *where* $EvI_{i,z}$ $(SubPd_{i,j}, Td_z)$ for $z = 1..n$ *and*
   ST_X509.Validity.notbefore $< Td_z <$ *ST_X509.Validity*.notafter

Solutions: P_i could check that the *ST_X509* will not be used more that once. The provider needs to store the *ST_X509.Issuer*, *ST_X509.Extensions.ID* and *ST_X509.Extensions.Tp* for each *ST_X509* received and next, compare them with the new ones received. Since *ST_X509.Extensions.ID* is the result of applying the hash to the purchases, intermediary and *ST_X509.Extensions.Tp* is a unique identifier of *ST_X509* for each *C*.

c) *Impersonating Intermediary*: A man in the middle (MITM) tries to impersonate the valid intermediary.

Solutions: There are two possible scenarios for this attack. First, when *C* sends the purchases and the *ST_X509*, the MITM tries to change the intermediary's identifier and manipulate the payment process. In our solution, the *ST_X509* includes the identifier of the intermediary in the field *ST_X509.Holder*. This identifier has the information of intermediary's identity certificate (*PKC*). As the *ST_X509* is signed by *C* it is impossible to modify this field. The real identity of the intermediary can be checked by his *PKC*.

The second attack is when the intermediary sends the short public key to *C* and the MITM tries to change this key. It can be avoided by sending this key signed by the intermediary.

3.3.3. *Other security discussions*

Using the intermediary PKC, it is possible to obtain the same security results as in our solution. However, we decide to use a temporal key for the following reasons:

- The authorization to the intermediary does not need to consider the lifetime of the PKC. If the intermediary needs to change his PKC, he does not need to stop the distribution

process, because the public key certificate of the intermediary is involved only in the initial process.
- Since this key has a short lifetime (associated to the authorization) for validating the evidence of intermediary participation (EvI_i) it is not necessary to check the CRL (certificate revocation list).
- This key could be shorter than the *PKC*, reducing the number of signatures in the distribution process.
- To include scenarios where the identity of the intermediary is not important or confidential for the provider. The intermediary identity can reveal some private info of *C* such as: geographic position in applications where the intermediary is a network access point (such as airport, subway, etc.), or preferences in application where the intermediary represents a specific web site.

4. Integrating multiparty payment with intermediary in 3D-Secure™ model

4.1. *Model payment of 3D-Secure™*

The three-Domain Secure (3-D Secure™) (3-D Secure) model of VISA provides the issuers with the ability to authenticate cardholders during an online purchase. This reduces the fraudulent use of credit cards and increases traceability of the transaction. The model divides the payment system into: Issuer Domain, Acquirer Domain and Interoperability Domain.

- The **issuer domain** is integrated by the Cardholder, a Visa member financial institution (Issuer) and a VISA component Access Control Server (ACS). This domain is responsible for managing the enrolment of their cardholders in the service and for authenticating cardholders during online purchases by means of ACS.
- The **Acquirer domain** is integrated by Merchant, a VISA financial institution (acquirer) and a VISA component Merchant Server Plug-in (MPI). This domain is responsible for defining the procedures to ensure that merchants participating in the Internet transactions are operating under a merchant agreement with the Acquirer, and providing the transaction processing for authenticated transactions by means of MPI.
- The **Interoperability Domain** is integrated by Visa Directory Server (DS) and Authentication History Server (AHS). The Visa directory Server handles all the communication between Merchant and the appropriate ACS in the process of request if the payment authentication is available. AHS stores the messages from the ACS for each attempted payment authentication and could be used by acquirers and issuers in case of disputes.

The following figure represents the Domain model of VISA and the principal flows in the payment protocol.

4.1.1. *The payment protocol*

Principal Messages
- **VEReq** – Message from MPI to the DS or from DS to the ACS, asking whether authentication is available for a particular card number
- **VERes** – Message from the ACS or the DS, telling the MPI whether authentication is available.
- **PAReq** – Message request sent from the MPI to the ACS (via the cardholder browser), to issuer to authenticate its cardholder.
- **PARes** – Message formatted, digitally signed, and sent from the ACS to the MPI (via the cardholder browser) providing the results of the issuer's 3-D Secure cardholder authentication

Flows of messages
1. First, the cardholder indicates the decision to buy, sending the purchases and payment info at this moment, MPI software is activated.
2. The MPI sends a message (**VEReq**) to the DS to determine whether authentication services are available for the cardholder.
   - If the cardholder is enrolled and authentication is available, the response message (**VERes**) instructs the MPI on how to contact the ACS (protocol continues with step 3).
   - If the account number of the cardholder falls outside of participating card ranges, the merchant proceeds with a standard authorization request.
3. The MPI sends an authentication request (**PAReq**) to the ACS. This is usually sent via the cardholder browser.
4. The ACS authenticates the cardholder by causing an authentication dialog to be displayed to the cardholder asking for a password, or by some other authentication method, such as a Visa chip card. The ACS formats and digitally signs the authentication response (**PARes**), then returns it to the MPI.
5. If the authentication response indicates successful authentication, the merchant forwards an authorization request with the requisite data to its acquirer for submission into an authorization system.

4.2. *Intermediary-3D Secure™*

We propose to adapt slightly the 3D Secure™ protocol to include our proposal. In this way it would be possible to take advantage of the intermediary role to decrease the number of customer operations (cardholder too) comparing with the possible solution of applying 3D Secure™ protocol for each involved providers. It also offers the possibility of making secure purchases with providers not enrolled in the VISA system. No modifications are proposed in the standard steps, nor in the messages of VISA but only in the flow of messages and the meaning of some fields in order to include the intermediary entity.

In our proposal (Fig. 3), the intermediary will be in charge of the cardholder authentications during online process (flow 1–4, described before) until the message **PARes** is received. The intermediary is a member of the 3D secure system and has installed the MPI for validating the cardholder. The IN guarantees that only one authentication dialog will be displayed to the cardholder, instead of one for each merchant. When the card validation ends, the intermediary makes the secure distribution payment (described in Section 3.2) to the providers. Next, providers check the received payment (signature of the intermediary and the customer). If

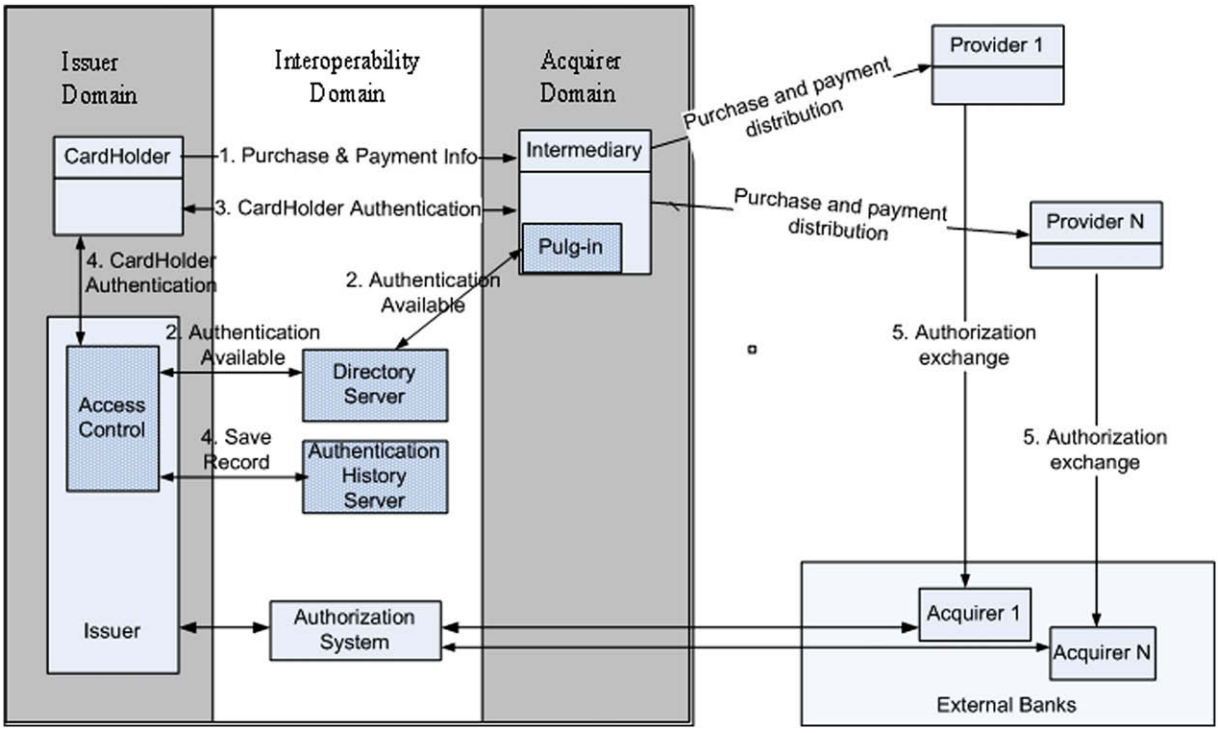

Fig. 3 – 3D Secure™ protocol with intermediary.

everything is correct, providers are able to make the Merchant's Authorization Exchange (flow 5) and conclude the payment operations with the issuer (using some authorization system within the network of banks).

In order to implement our approach, it is only necessary to modify some fields (relative to the merchant and the purchase information) in the messages **VEReq**, **PAReq**. The intermediary does not have an acquirer (*VEReq.Acquired BIN*) and acquirer-defined merchant identifier (*VEReq.Merchant ID*) because it does not receive the final payment.

Since in our model the intermediary does not receive the final payment, it does not have an acquirer (*Acquired BIN*) or acquirer-defined merchant identifier (*Merchant ID*).However, the intermediary requires that the 3D Secure model assigns an internal code to separate the intermediary entity and final merchant from the VISA financial institution (traditional model).

We propose to modify the fields of information about the merchant to information about the intermediary:

VEReq
Acquirer BIN:
  Acquiring institution identification code (*In the standard*)
  VISA identification code for the intermediary (*In our proposal*)
Merchant ID
  Acquirer-defined merchant identifier (*In the standard*)
  VISA identification code for the intermediary (*In our proposal*)

PAReq
Acquirer BIN:
  From **VEReq**
Merchant ID
  From **VEReq**
Merchant Name
  Merchant name on Authentication Request Page (*In the standard*)
  Intermediary name on Authentication Request Page (*In our proposal*)
Merchant Country Code
  Country Code of the Merchant (*In the standard*)
  Country Code of the Intermediary (*In our proposal*)
Merchant URL
  Fully qualified URL of merchant website (*In the standard*)
  Fully qualified URL of intermediary website (*In our proposal*)

The fields with information about the purchases must be also adapted in order to represent the total amount (*PurchaseAmount*) and to describe all the order with products and providers (*Order description*). The *Order description* field will be used by the issuer for checking the request of payment from the providers.

PAReq
Purchase Amount
  Total amount of all purchase (items through different providers)
Order Description
  Brief description of items purchased. (*In the standard*)
  This field is optional. In our solution we need to change it to ''required field'' in order to store purchase evidences organized by providers. This means Pi, $SubPd_{i,j}$ (*In our proposal*)

Finally, due to the fact, that many providers are involved, the ACS needs to generate many $\mathbf{PARes_i}$ for each $P_i \in P'$.

The message flows of the Intermediary 3D Secure must been modified in the following way:

1. *Similar to the standard,* except that the cardholder indicates the decision to buy from different providers. Also, it must be included the short time certificate (*ST-X509*).
2. *3 y 4. Similar to standard* except that the modifications of the messages **VEReq, PAReq** and the new $\mathbf{PARes_i}$
5. *In our proposal* the MPI validates all the $\mathbf{PARes_i}$ and next, if the authentication is successful, the Intermediary distributes the appropriate list of products $SubPd_{i,j}$, and the authentication response $\mathbf{PARes_i}$ to each providers P_i. Also in this message, it must be included the *ST_x509* and the intermediary digital signature EvI_i as his evidence of participation.
6. Upon reception, each providers P_i checks the cardholder's signature over *ST-X509*, the integrity of the purchase, the **PARes** and, finally, each provider finishes the payment process forwarding the authorization to the acquirer.

5. Analysis and discussion of our proposal

5.1. *Advantages of the intermediary-based model*

In general, the customer device is the participating entity with less computational resources. This situation has usually motivated the shift of weight operations towards other entities (or the decrease of the number of these ones) without lost of security features. Our proposed intermediary-based model makes it possible: the customer is able to simultaneously pay to multiple providers without the necessity of visiting several *checkout* web pages and entering the corresponding personal/payment information with the goal of completing the purchase. By means of our intermediary-based model, the customer would make a single payment transaction through the intermediary, which would afterwards distribute it. This strategy clearly decreases the number of customer interactions, simplifies the client-side application and permits the centralization of the communication with any payment system (PS) through the intermediary entity.

Taking as reference our model, the providers would not have to care about the setting-up, configuration and maintenance of a diversity of merchant-side plug-ins that correspond to each payment system. The providers receive/send all the messages of payment transaction through the intermediary entity. Once adopted, our model allows merchants to implement current secure payment systems, or even future ones, without the need of modifying their back-end applications. As result of all these advantages, costs are minimized. This circumstance specially favors to modest providers, as well as, small and medium-size companies to be open to e-commerce.

Comparing our intermediary model and secure proposal with some previous work (see Table 1) we can conclude that our solution has better features and less security restrictions.

Unlike solutions such as 1, 2, 3 and 7, the appearance of intermediary *IN* in our model decreases the number of interactions of the customer and the number of operations in the client application, by centralizing the payment in scenarios with many providers. Moreover, our secure solution does not require a TTP to protect the transactions (such as 1, 2 and 4), as well as, it does not assume a trustworthy *IN* (such as in 1, 2, 3, 4 and 6). Finally, although we require a Public Key infrastructure in order to authenticate *IN*, C and merchants, this requirement is easier to implement than the coordination between *k* providers (as suggested in 5). Public key certificates and their infrastructures utilities are widely used nowadays in many communications protocols (e.g. SSL-based protocols).

Table 1 – Comparing our intermediary model.

	Multiples providers	TTP	Customer operations	Restrictions
1) OPELIX	NO	YES	MANY	–
2) LITESET/A++	NO	YES	MANY	**trusted** assumption with customer
3) COYOTE	YES	NO	MANY	The IN is a **TTP**
4) LITESET/A++-multi	SI	YES	FEW	**trusted** assumption with customer
5) MultiSET	YES	NO	FEW	*k* providers together
6) P2P_IN	YES	NO	MANY	**trusted** assumption with provider
7) Proxy signature	NO	NO	MANY	–
8) Our model	YES	NO	FEW	PKI

5.1.1. *Advantages of the new Intermediary-3D Secure™ protocol*

As we show above in this paper, the integration of the intermediary-based model with the 3D Secure™ system is feasible and provides many advantages. In the following paragraphs, we compare our Intermediary-3D Secure™ protocol for many providers with the closed solution that is the way in which currently occurs the multiparty payment transactions, i.e. *NP* iterations of the original 3D Secure™ protocol, where *NP* is the number of providers. We focused on this comparison in the number of client side operations to determine the *transaction cost*.

In Table 2, we point the number of main customer operations for each provider in the current 3D Secure™ scheme versus our proposed Intermediary-3D Secure™. Consider the following notations:

- *NP*, number of providers;
- *cSSL*(*Y*), number of operations to establish one SSL connection with the entity *Y*;
- *cACS*, number of operations with the ACS;
- *AI*, authentication info (e.g. password, pin) to be checked by ACS;
- *nDS*(*X*), number of digital signatures over message *X*;
- (*bytes*)*X*, amount of bytes of the message *X*.

- ***SSL connections***: It refers to the number of operations required to establish SSL connections with all the involved entities. 3D Secure™ protocol requires client side application to establish SSL connections with each involved provider, and one connection with the ACS. Our solution

Table 2 – Transaction cost comparison between 3D Secure and Intermediary-3D Secure in multipayment scenarios.

	3D Secure™ with many providers		Intermediary-3D Secure™
SSL connections	$NP^{*}cSSL(P) + NP^{*}cSSL(\mathrm{ACS})$	$\gg$	$1^{*}cSSL(IN) + 1^{*}cSSL(\mathrm{ACS})$
Digital signature	**0**	$<$	$nDS(ST\_X509)$
Bytes transmission	$NP^{*}(bytes)PAN + NP^{*}(bytes)SubPd_{i,,j} + NP^{*}(bytes)PAReq + NP^{*}(bytes)ST\text{-}X509 + NP^{*}PARes$	$\gg$	$(bytes)PAN + (bytes)\,SubPd_{i,j} + (bytes)PAReq + (bytes)ST\text{-}509 + NP^{*}PARes_i$

reduces so many SSL connections to only one to the *IN* and only one connection to the ACS.

- ***Bytes transmission:*** Amount of sent bytes through the communication channel, from the client side application. In the original 3D Secure™ protocol, the cardholder application sends the authentication info, payment order, **PAReq**, and **PARes** for each participating provider. Nevertheless according to our extension, the cardholder application just needs to send: PAN and $SubPd_{i,j}$ to the *IN*, the **PAReq** to ACS and authentication info to the ACS, but finally he need to send the $\mathbf{PARes_i}$ for each involved providers P_i.
- ***Digital signature:*** Number of required digital signatures that the client application (with the participation of cardholder) should perform. In the 3D Secure™ protocol, the authentication guarantees are directly provided by the ACS. However, in our version we need to add a digital signature over the *ST_X509* certificate, in order to provide purchase results integrity to the providers.

According to Table 2, we can conclude that our proposal improves the traditional 3D Secure™ protocol reducing the operations need for multipayment. Although the cardholder needs to perform a digital signature operation, the number of SSL connections and amount of byte transmission through the communication channel are more significant. Furthermore, in some situations (for example m-commerce applications) these operations are more expensive because the connections have higher cost, and also, the costs of the connections depend on the amount of bytes sent.

5.2. *Application in real scenarios*

The possibility of managing payments with multiple providers is especially interesting for different kind of brokerage business models such as the virtual malls or marketplaces based on C2B models. This is a hosting service for online providers that charges setup, monthly listing, and/or transaction fees (an example isAmanzon.com). By adding the multi-payment functionality, it allows us to include services for buying a set of products from different providers during the same transaction. In such a case, the purchase process that includes web searching, management of different payment methods, etc. is simplified. But note that, afterwards these applications simply redirect the payment process to the involved providers.

On the other hand, the brokerage model is represented by the well-known **search agents**. These ones are software agents or ''robots'' used to search-out the price and availability for a good/service specified by the buyer. Typical examples are: products search agents as shopping.yahoo, music search service as Pandora, etc. Nevertheless, in all of these cases, the agents do not manage multipayment transactions. Therefore, adding the possibility of selecting multiple products and paying for them in a single transaction could increase the value of these business models.

The Marketplace Exchanges (e-Marketplace) based on B2B brokerage models, (e.g. Chemconnect for chemical industry exchange, www.chemconnect.com), could also take advantage of our payment intermediary model. This scheme connects buyers and sellers in order to exchange a full range of services, covering from market assessment to negotiation and fulfillment. Customers in this model generally place orders that include products by more than one provider (a special case is supply aggregation). By integrating the multipayment functionality of our intermediary model in these schemes, the number of customers transactions might be reduced, and consequently the impact in periodical purchases would not be negligible.

6. Conclusions

Security is a critical point in the implementation of new business models. The payment is the process with the highest need for security because it is where the customer legally ends the business by making the funds transfer. Electronic payment solutions are mostly focused on traditional two-party business models. However, many business models involve some intermediary entities to help negotiation.

In this paper, we analyze a multiparty electronic commerce model in which an intermediary plays the role of payment mediator between one customer and many providers. Here, the customer delegates the multipayment transactions to the intermediary and creates a single secure transaction between customer and his providers. We propose a secure solution in which the customer creates a short-term certificate for the intermediary as authorization credential to forward and distribute the payment info. This will be used in the distribution process to create evidence of the intermediary's participation. Also by this means, the provider can obtain the customer's authentication and assurance of purchase integrity. Unlike to other secure solutions in e-commerce models with intermediary, in our secure solution the intermediary is not represented as a trusted entity (is not a TTP).

As result of this research, we adapt 3D Secure™ protocol with the goal of including our intermediary-based model. This takes advantage of the intermediary features to offer the possibility of making secure purchases with providers that are not enrolled in the *Verified by Visa* system, by means of minor adaptations.

After analyzing our protocol for secure multipayments, we demonstrate that the transactions cost is minimized in the client side, for these scenarios with multiple vendors.

We can conclude that our intermediary model provides benefits both customers and providers, and it could improve the attractiveness of the traditional business models. In future works, we aim to involve a payment chip card in the transactions of the 3D Secure™ protocol with multiple providers, and to adapt the protocol to voice channel 3D Secure™ solutions.

REFERENCES

3-D Secure. System Overview. 70015–01 External version 1.0.2 May 01, 2003. Copyright 2002–2003 Visa International, http://partnernetwork.visa.com/pf/3dsec/download/trk_3dsec_system_overview_v102.pdf.

Bella G, Bistarelli S. Information assurance for security protocols. Computers and Security 2004;24:322–33. Elsevier.

Carbonell M, Onieva J, Lopez J, Zhou J, Galpert D. Simulation model for the estimation of timeouts in non-repudiation protocols. In: ICCSA Workshop on Internet Communications Security, LNCS 3043, May 2004. p. 903–14.

Chen T. An English auction scheme in the online transaction environment. Computers and Security 2004;23:389–99. Elsevier.

Gou L, Wang G. Insider attacks on multi-proxy multi-signature schemes. Computers and Electrical Engineering 2007;33(2): 88–93.

Hauswirth M. Jazayeri M, Schneider M. A phase model for e-commerce business models and its application to security assessment. Project OPELIX. In: Proceedings of the Hawaii international conference on system sciences, Maui, Hawaii; 3–6 January 2001.

Jenamani M, Mohapatra P, Ghose S. A stochastic model of e-customer behavior. Electronic Commerce Research and Applications 2003;2(1):81–94.

Kim S, Lee W. A payword-based micropayment protocol supporting multiple payment. In: Proceedings of The 12th international conference on computer communications and networks; 20–22 October 2003. p. 609–12.

Lau R. Towards a web services and intelligent agents-based negotiation system for B2B eCommerce. Electronic Commerce Research and Applications 2006;14 [accessed 14.07.06].

Lee Z, Jun Lee S. The effect of buyer feedback scores on internet auction prices. Journal of Organizational Computing and Electronic Commerce 2006;16(1):51–64.

Li H, Su S, Lam H. On automated e-business negotiations: goal, policy, strategy, and plans of decision and action. Journal of Organizational Computing and Electronic Commerce 2006; 16(1):1–29.

Lia L, Tzenga S, Hwang M. Generalization of proxy signature-based on discrete logarithms. Computers and Security 2003; 22(3):245–55.

Liberty Alliance Project. Liberty alliance contractual framework outline for circles of trust, http://www.projectliberty.org/liberty/files/whitepapers/liberty_alliance_contractual_framework_outline_for_circles_of_trust; 2007 [accessed 06.03.07].

Lin F, Lin Y. Integrating multi-agent negotiation to resolve constraints in fulfilling supply chain orders. Electronic Commerce Research and Applications 2006;5(4):313–22.

Onieva J, Zhou J, Lopez J. Practical service charge for P2P content distribution. In: International conference on information and communications security, LNCS 2836. Huhehaote, China: Springer, ISBN 3-540-20150-5; 2003. p. 112–23.

Onieva J, Zhou J, Lopez J, Carbonell M. Agent-mediated non-repudiation protocols. Electronic Commerce Research and Applications 2004:152–62. Elsevier.

OPELIX Project Reference: IST-1999-10288. An open personalized electronic information commerce system, www.opelix.org. Start Date: 2000-01-01, End Date: 2002-02-28.

Rappa M. The utility business model and the future of computing services. IBM Systems Journal(1), http://digitalenterprise.org/models/models.html, 2004;43.

Sans O, Agnew G. An efficient multiple merchants payment protocol for secure electronic transaction based on purchase consolidation'' ISEC 2001. In: LNCS, 2040. Berlin Heidelberg: Springer-Verlag; 2001. p. 1–19.

Sen S, Saha S, Hernandez K. Buyer agent to enhance consumer awareness: SAATHI. Electronic Commerce Research and Applications, Summer 2007;6(2):209–18.

Téllez J, Sierra J. An anonymous account-based mobile payment protocol for a restricted connectivity scenario2. In: DEXA workshops 2007; 2007. p. 688–92.

Tsai W, Paul R, Song W, Zhibin C. Coyote: an XML-based framework for web services testing. In: Proceedings. 7th IEEE international symposium on high assurance systems engineering; 23–25 October 2002. p. 173–4.

Tsiakis T, Sthephanides G. The concept of security and trust in electronic payments. Computers and Security 2005;24(1):10–5.

Wang Y, Li T. LITESET/A++: a new agent-assisted secure payment protocol. In: IEEE international conference on e-commerce technology (CEC'04); 2004. p. 244–51.

Wang Y, Varadharajan V. A mobile autonomous agent-based secure payment protocol supporting multiple payments. In: IEEE/WIC/ACM international conference on intelligent agent technology; 2005. p. 88–94.

Wang Y, Tan K, Ren J. PumaMart: a parallel and autonomous agents based internet marketplace. Electronic Commerce Research and Applications 2004;3(3):294–310.

Wang Z, Qian H, Li Z. Hybrid proxy multisignature: a new type multi-party signature. Information Sciences 2007;177(24): 5638–50.

Yarom I, Rosens J, Goldman C. The role of middle-agents in electronic commerce. Ieee Intelligent Systems and Their Applications 2003;18(6). ISSN: 1541-1672:15–21.

Yu Y, Xu C, Huang X, Mu Y. An efficient anonymous proxy signature scheme with provable security. Computer Standards and Interfaces 2008;17 [accessed 17.05.08].

Mildrey Carbonell is Assistant Professor at the Computer Science Department of the University Carlos III de Madrid. She is M.Sc. in Computer Science and now she is Ph.D student in Computer Science. She is a member of the Security Group and her research activity is focused in the security requirements (non-repudiation and authentication fundamentally) of payment protocols for new applications scenarios (such as mobile, disconnected, intermediaries and other e-commerce models). She has participated in numerous research projects, UBISEC, CASENET, ASPECT-M, and she has published articles in journals and proceedings related with the security and electronic commerce. At the present, she works as Security Consulter in SATEC (www.satec.es).

Jose M. Sierra is a Assistant Professor at the Computer Science Department of the University Carlos III of Madrid. He is Ph.D. in Computer Science and M.Sc. in Business Administration. His research work is centered in the area of the Internet Security and, in this area, at the present time he is working and researching. He has participated in

numerous research projects and has published articles in journals related with the Security in the Information and Communication Technologies.

Javier Lopez received his M.S. and Ph.D. in Computer Science in 1992 and 2000, from the University of Malaga, respectively. From 1991 to 1994, he worked as a System Analyst in the private sector and in 1994 he joined the Computer Science Department at the University of Malaga as an Assistant Professor, where he actually is an Associate Professor. His research activities are mainly focused on information and network security, leading some national and international research projects in those areas. Prof. Lopez is the Spanish representative of the IFIP TC-11 (Security and Protection in Information Systems) Working Group. Also, he is Co-Editor in Chief of the International Journal of Information Security, member of the Editorial Boards of Information Management and Computer Security Journal and International Journal of Internet Technology and Secured Transactions, and member of the Steering Committee of ERCIM's Working Group on Security.