Comparing and contrasting micro-payment models for E-commerce systems

Xiaoling Dai [1], John Grundy [1] and Bruce W N Lo [2]

[1] Department of Computer Science
University of Auckland
Private Bag 92019, Auckland, New Zealand
xiaolingd@hotmail.com, John-g@cs.auckland.ac.nz

[2] School of Multimedia Information Technology
Southern Cross University
PO Box 157, Lismore, NSW 2480, Australia
blo@scu.edu.au

Abstract

Current macro-payment systems used by most E-commerce sites are not suitable for high-volume, low-cost produce or service purpose, such as charging per-page for web site browsing. These payment technologies suffer from use of heavy-weight encryption technologies and reliance on always on-line authorisation servers. Micro-payment systems offer an alternative strategy of pay-as-you-go charging, even for very low cost, very high-volume charging. However, several different micro-payment schemes exist, not all suitable for all E-commerce uses. We compare and contrast several micro-payment models and outline a new micro-payment technology we have been developing.

Keywords: electronic commerce, micropayment, off-line.

Introduction

Macro-payment systems are used by most E-commerce systems today. These typically use credit card debiting, digital cash or real-time bank transfers, where a customer pays for products or services before or at the time of delivery. Such systems typically use complex encryption technologies and require communications with an authorisation server to request and confirm payment. This model suits low-to-medium volume transactions of medium-to-high value e.g. books, food, office stationary, home appliances, toys and so on.

There is a trend towards charging for site content on the Internet [2] in order for companies to make direct profits from information they provide, rather than relying on fickle or insufficient on-line advertising revenue [9, 10]. For example, many sites have become subscription-only access e.g. on-line newspapers, academic and trade periodicals, help and advice columns, and so on. Subscription has the disadvantage of locking customers to one site (they need to subscribe to every site they want access to) and a "one size fits all" scenario where even if the customer wants a few items from the site, they have to pay for them all.

An alternative model is where a customer pays as they go from a previously acquired (by macro-payment) E-wallet with E-coins i.e. is charged per-page or per-group or per-download for material, often very low cost per item [7, 9, 10]. Ideally they can move to other sites and use the same E-money. This is the micro-payment model of on-line information, product and service purchase.

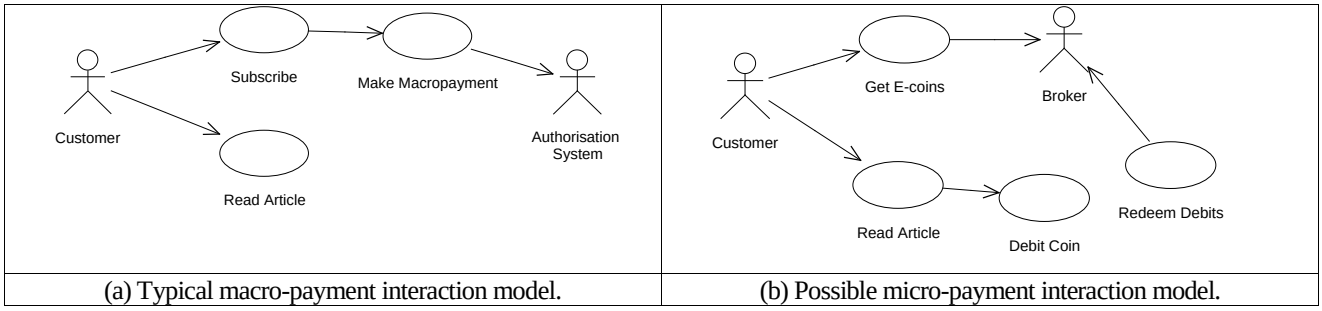
In the following sections we introduce an example scenario – an on-line newspaper – that wants to charge on an article usage basis. We review the typical macro-payment model's interactions between customer, vendor (E-newspaper site) and authoriser (bank or credit-card company). We then compare several micro-payment models and discuss their various advantages and disadvantages for supporting this pay-as-you-go purchasing model.

Motivation

Assume a reader wants to read an on-line newspaper. Using subscription-based payment, they would first have to subscribe to the newspaper by supplying payment details (credit card etc) and the newspaper system would make an electronic debit to pay for their subscription, by communicating with an authorisation server. The user would then normally go to the newspaper's site where they login with an assigned user name and password. The newspaper looks up their details and provides them access to the current edition if their subscription is still current. If the user's subscription has run out, they must renew this by authorising a payment from their credit card. Figure 1 (a) outlines the key interaction use cases for this scenario. Problems with this approach are that there is no anonymity for the user (the newspaper system knows exactly who they are and when and what they read), they can not browse other newspapers without first subscribing to them too, and they must pay for the whole newspaper, even if they want just one or two sections or articles.

An alternative approach is a micro-payment model. The user first goes to a broker and purchases "E-coins" using a single macro-payment. These are stored in an E-wallet on the user's machine. The user can then visit any newspaper site they wish, their wallet giving the site an E-coin. Each time they view an article (or section or page, depending on the item charged for) their E-coin is debited. The vendor redeems debits with the broker (for "real" money) periodically e.g. each night/week. The user can move to another site and unspent money associated with their E-coin is transferred from the first vendor to the second. If coins run out, the

user communicates with the broker and authorises another macro-payment debit. Figure 1 (b) outlines the key interaction use cases for this scenario.



(a) Typical macro-payment interaction model. (b) Possible micro-payment interaction model.

Figure 1. Two on-line newspaper interaction scenarios.

The standard macropayment methods cannot be effectively or efficiently applied for buying inexpensive information goods, like single articles of an on-line newspaper, because transaction costs are too high. Encryption mechanisms used are slow and each transaction typically "costs" a few cents. Macro-payment suits spending small numbers of large amounts. An Internet micropayment system would allow spending large numbers of small amounts of money at web sites in exchange for various content or services, as in the E-newspaper scenario above. The design of micro-payment systems are usually quite different from existing macro-payment systems, since micropayment systems must be very simple, secure, and efficient, with a very low cost per transaction. This must also be taken into consideration for transaction security: high security leads to high costs and computation time. For micropayments low security can be applied.

There are a number of payment systems in various stages of development from proposals in an academic literature to systems currently in commercial trials. Payment protocols that are exclusively designed for electronic payments in a normal customer to vendor transaction can be categorized as either online or offline protocols. On-line payment systems include *iKP*, *Netbill*, and *CyberCas*h [1,4,15]. In on-line systems, every payment needs to be authorized by the central payment authority that issued the coin in order to prevent double spending. This is called an on-line payment scheme since the issuing bank is involved in every transaction. Unfortunately the central organization very quickly becomes a potential bottle-neck and point-of-failure.

Protocols that do not rely on a third party (broker) to guard against double spending are called off-line micropayment protocols, such as *PayWord, MiniPay* and *NetPay* [13,8,6]. These protocols are typically credit based. In some there is no protection mechanism to prevent a customer from double spending, and spending more than the balance in their account (overspending). Double spending is detected at the time of the clearing process, when the vendor turns in the received coins to their respective banks. Once double spending is detected, the malicious customer are usually penalized and expelled. Though off-line protocols have received a lot of attention from researchers and cryptographers, no off-line payment systems yet exist in the general public use.

Macro-payment Model

Within macropayment systems there are three distinct payment methods that are credit card based, digital cash, and electronic check (account based). Credit based payment systems, such as SET and CyberCash [14,15], are both online and post paid payment by credit card. There are many payment systems based on e-cash payment, such as DigiCash which is online and prepaid payment system, and CAFÉ which is offline and prepaid payment system [5,3]. NetCheque employs an account server to transfer and authenticate electronic checks [12]. Figure 2 represents a model how credit cards can be used in a secure way across the Internet. The protocol is called CyberCash that provides customer software, vendor software and a gateway to support the secure communication of credit card transaction over the Internet.

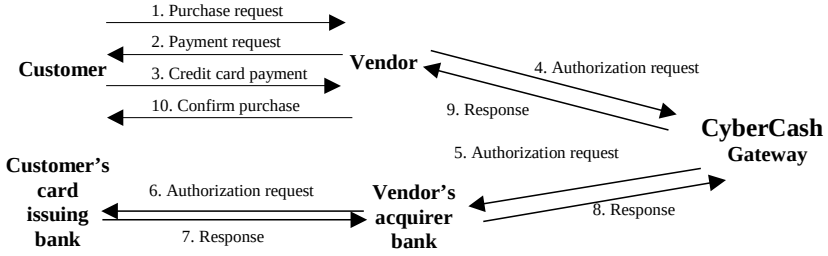


Figure 2. Typical macro-payment interaction model.

Macropaymet systems provide high level security, but all on-line payment schemes introduce a central bottleneck, a single point of failure that increases payment latency. It also raises the cost of the transaction, and imposes a minimum cost per transaction, as the bank is faced with the real cost of authorizing each transaction. As a result, the macropayment systems are not suitable for high-volume, low value payment transactions.

Micro-Payment Models

We review the key concepts of several micro-payment systems below, identifying their key strengths and weaknesses.

4.1. Millicent

Millicent, a micropayment system implemented by Digital Equipment Corp, now owned by Compaq, went live in June 1999 in Japan, with wallets starting at 1000 yen and payments as small 5 yen [11]. Millicent does not fall into either the online or the offline category, but rather is a distributed allocation of funds to vendors, who locally authorize payments. Millicent introduces a new kind of currency - scrip, which is digital money that is issued by a single vendor. Scrip has a value, just as cash does, but it has value only when spent with specific vendor. Scrip consists of a signed message attesting that a particular serial number holds a particular value. In addition to the necessary contents of electronic cash the scrip will also hold an expiration date and information on the particular vendor with whom the scrip can be redeemed. Figure 3 shows key interactions.

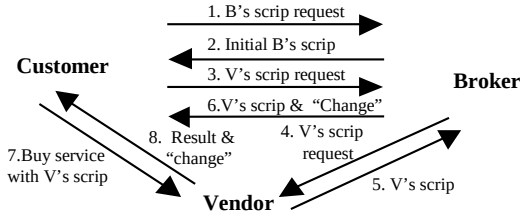


Figure 3. Millicent interactions.

- *Broker scrip request:* The customer buys the broker scrip at start of the day and the broker returns initial broker scrip and associated secret.
- *Vendor scrip request:* The customer requests vendor scrip paying with broker scrip from the broker. There are three models in which the broker gets the vendor scrip. *Scrip warehouse model:* the broker requires the scrip directly from the vendor. *Licensed scrip production model:* if a broker buys a lot of scrip for a specific vendor, the vendor sells the right to the broker to generate vendor scrip that the vendor can validate and accept. *Multiple brokers' model:* customer's broker needs to contact vendor's broker to buy the scrip. After the broker gets a vendor's scrip, the broker sends the scrip, associated secret and "change" broker scrip to the customer.
- *Transaction:* The customer buys services with vendor scrip and the vendor returns information goods & "change". The customer continues using the change to make more purchases with this particular vendor. When the customer wants to purchase with another new vendor, he needs to request new vendor scrip.

Millicent uses no public-key cryptography and is optimized for repeated micropayments to the same vendor. Its distributed approach allows a payment to be validated, and double spending prevented without the overhead of contacting the broker on-line during purchase. Key drawbacks with Millicent include: the broker must be on-line whenever the customer wishes to interact with the new vendor; the customer must nearly always be able to connect to the broker in order to be sure of the ability to make payments; the vendor scrip is vendor-specific and has no value to another vendor; and transactions are very complex when the customer and the vendor have different brokers.

4.2. Mpay

This micropayment system proposal from IBM was previously named MiniPay [8]. Mpay is very similar to the billing mechanism of the third party value added services of the phone networks. The customer deals with his issuer, the vendor deals with his acquirer and the issuer and the acquirer settles the accounts. The system is suitable for selling inexpensive information and other similar services that are usually delivered on-line. Figure 4 shows key Mpay interactions.

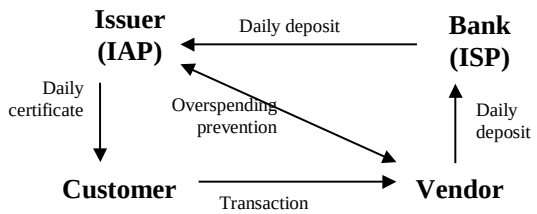


Figure 4. Mpay interactions.

- *Daily certificate request:* In the Mpay system, the customer connects every day to his issuer to receive a daily certificate. This certificate signed by the issuer whose public key is known by all vendors. The certificate states that the customer has an account and tells the recommended offline limit for the daily purchases.
- *Transaction (online):* The customer clicks on a specific type of a tag in his browser. The system encodes the cost, daily certificate and other necessary information and sends to the server. This enables the customer to send the payment at same time as the query for the vendor. The payment order is piggybacked on the request. The vendor verifies the signature

of the issuer on the certificate. If the daily limit is not exceeded, the vendor immediately responds to the request. However if the daily limit would be exceeded, the vendor connects to the issuer by sending extra spending request. The issuer will or will not send an extra spending reply information to the vendor according to the customer's record.

- *Daily deposit:* At a fixed period the vendor sends all the payment orders from all customers in a single, signed deposit message to the bank. The bank sorts the payment orders based on the issuer of each of the customers and sends the single signed deposit message to them.
- *Daily process:* At the end of the day or at the first purchase the next day, the customer contacts the issuer for their daily process. In this process the customer and the issuer compare their records for the previous day, all matching records are erased and replace them with a summarized and signed document of purchases and the balance.

Mpay is based on a notational model and has off-line capability in its daily certificate. Mpay only uses one or no public key operation per purchase, so the transaction cost is low. There is no extra communication required in the system, because of the payment order piggybacked on the information request. It is a real 'pay per click' system, the customer can ease to use it. However the major shortcoming of the system is that the customer can pay nothing to the issuer who still needs to pay the bank after purchasing goods. The issuer can protect itself by requiring a deposit from the customer, and by terminating Mpay and Internet access, but the criminal is still free to spend for a full day. In a worldwide there are billions of online customers to use the payment system and a lot of issuers, it seems to be impossible to terminate Mpay and Internet access to such customers. Even though the termination of Mpay and Internet access is possible to some customers, this increases initialization cost, thus finally driving up the cost of the system. Furthermore, the protocol is not fully anonymous due to the after the fact policing requirements. Thus the issuer is able to collect a complete purchase profile of their customers.

4.3. PayWord

PayWord micropayment protocol proposed by Ron Rivest (MIT Laboratory for Computer Science, MA, USA) in 1996 [13]. The protocol aims to reduce the number of public key operations required per payment by using hash functions, which are faster. In PayWord customers generate their own "coins," or paywords, which are sent to vendors and then verified by brokers. Figure 5 shows key PayWord interactions.

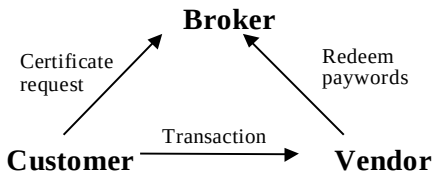


Figure 5. PayWord interactions.

- *PayWord certificate request:* In the beginning of the transaction, the customer establishes an account with a broker, who issues a digitally signed PayWord certificate, which contains identity and public key of the customer and other information. The certificate authorizes the customer to make PayWord chain (e-coins) and ensures vendors that the customer's paywords are redeemable by the broker.
- *Transaction*: When a user wishes to make a purchase at a vendor for the first time in a day, he first randomly picks a payword seed wn. The customer then computes a payword chain by repeatedly hashing wn: $wi-1 = h(wi)$, where $i = 1, \cdots, n$. The customer then sends the digitally signed commitment which includes w0 the root of the payword chair and the certificate to the vendor. It is used to show the customer's intentions of spending paywords there. To make m cents payment, the customer sends w1 through wm where m is the number of the paywords the customer wish to spend and the requirement of the information goods to the vendor. The vendor can easily verify this chain by hashing wm m times until he reaches w0. The vendor sends the information goods to the customer.
- *Redeeming:* At the end of each day, the vendor sends the customer's commitment and the highest payword spent to the broker. The broker verifies the paywords using the root w_0 and the customer's signature. If they are valid, the broker debits the spent amount from the customer's account and pays the vendor.

PayWord is an off-line system. The customer only needs to contact the broker at the beginning of each certificate lifetime in order to obtain a new-signed certificate. The system aims to minimize the number of public key operations required per payment using hash operations instead whenever possible. It is credit-based scheme where a user's account is not debited until some time after purchases. This provides more opportunity for fraud since a large number of purchases can be made against an account with insufficient funds. The e-coin (paywords) in the system is customer and vendor specific and the paywords in the chain have no value to another vendor.

4.4. NetPay

We present a new protocol called NetPay that allows customers to purchase information from vendors on the WWW [6]. NetPay, a secure, cheap, widely available, and debit-based protocol of a micropayment system, will be introduced for the WWW. NetPay differs from previous protocols in the following aspects: NetPay uses touchstones signed by the broker and Index's signed by vendors passed from vendor to vendor. The signed touchstone is used for vendor to verify

the electronic currency – paywords, and signed Index is used to prevent double spending from customers and to resolute dispute between vendors. There are no customer trusts required. Figure 6 shows key NetPay interactions.

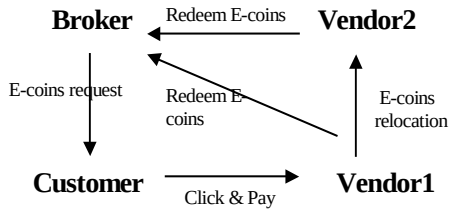


Figure 6. NetPay interactions.

- *E-coins request*: Before a customer asks for service from the first vendor V_1, he has to send a message which includes an integer n, the number of paywords in a payword chain the customer applied for and IDc to the broker. The broker completes two actions: (1) Debits money from the account of C and creates a payword chain which is same as PayWord. The customer only receives paywords $W1, W2, \cdots, Wn$ that are encrypted by customer's public key from the broker. (2) Computes the touchstone T which includes IDc and W_0 for that chain. T is signed by broker. This touchstone authorizes V_1 to verify the paywords using root W_0 and redeems the paywords with the broker.
- *Transaction:* When a customer makes a purchase from V_1, he sends a message which includes IDc, Payment $P = \{(Wj, j), (Wj+1, j+1), \cdots, (Wj+m-1, j+m-1)\}$ (m cents) and an order to the V_1. V_1 verifies the payment. If the payment P is valid, P will be stored for a later offline transaction with the broker, and the customer is supplied the information goods. If the paywords are stolen by an attacker, he only can spend the paywords in P to V_1. Multiple payments can be charged against the length of the payword chain, until the payword chain is fully spent or the customer no longer requires information goods on WWW.
- *Paywords Relocation:* When a customer wishes to make a purchase at V_2, he sends IP address of V_1, IDc, payment, and order to V_2. V_2 transmits IDc and IDv_2 to V_1 in order to ask for the Index. Then V_1 signs the $Index = \{IDv1, IDv2, I\}$ where I is the index of the last payword V_1 received along with the payword chain touchstone, and transmits them to V_2. The Index may be used for disputes between the vendors. V_1 verifies the payment using Index and W_0. If the payment is valid, it will be stored for a later offline transaction with the broker, and the customer is supplied the information goods. This transaction has two advantages: firstly, the transfer of the message from V1 to V2 does not involve the broker, it reduces the communication burden of the broker; secondly, the message includes the index of the paywords, it prevents the customer from double spending when the customer purchases from another vendor.
- *Offline Redeem processing:* At the end of each day (or other suitable period), for each chain, the vendor must send the touchstone IDc, IDv, and payment to the broker. The broker needs to verify each payword received from the vendor by performing hashes on it and counting the amount of paywords. If all the paywords are valid, the broker deposits the amount to the vendor's account.

NetPay is a basic offline protocol suitable for micropayments in distribute systems on the WWW. Since only the broker knows the mapping between the pseudonyms (IDc) and the true identity of a customer, the protocol protects the customer's privacy. The protocol prevents customers from double spending and any internal and external adversaries from forging, so it satisfies the requirements of security that a micropayment system should have. The protocol is "cheap" since it just involves small number public-key operations per purchase. NetPay can easily handle more transactions. In extended NetPay system, a coin can be divided in small denominations, i.e. it has divisibility. NetPay is extremely powerful for a customer performing many purchases from a vendor, then change to another vendor.

Discussion

In Table 1 we compare these various E-commerce payment system models. We consider micropayment systems from the perspective of reads of the on-line newspaper (Customers), newspaper vendors, and mico-payment brokers or macro-payment authorisers.

The evaluation criteria we use include:

- *Ease of use*. The system must be easy to use for the customer. There is no login and PIN number required all time. The customer only needs to click and to buy a page in the web page with a micropayment system in a few seconds.
- *Security*. The aim of security in the payment protocols is to prevent any party from cheating the system. For customers and external adversaries the forms of cheating security, which are specific to payment schemes, are double spending of coins and creation of false coins forgery during payment.
- *Anonymity*. The customer anonymity should be protected. A fundamental property of physical cash is that the relationship between customers and their purchases is untraceable. This means that the payment systems do not allow payments to be traced without compromising the system's security. This may encourage some potential customers to start using the payment system.
- *Multi currency*. The micropayment systems should be able to operate with multiple currencies, by converting the currencies either inside or outside the system.

- *Divisibility* – the protocol supports multiple denominations and a range of payment values.
- *Performance* – the protocol provides high-volume payment support.
- *Robustness* – the protocol is tolerant of network bottlenecks and broker/authoriser down-time.

System/ property	CyberCash	Millicent	Mpay	PayWord	NetPay
Ease of use	**Low,** Customer contacts Broker every transaction.	**Medium,** Customer nearly always contacts Broker.	**High,** Customer only needs to click and see what he pays.	**Medium,** Customer generates and manages e-coins for every Vendor.	**High,** Customer clicks and gets the content.
Security	**High,** the system performs checking, clearing and recording of transactions.	**Medium+,** the system prevents double spending by using Vendor-specific scrip.	**Medium,** the criminal is free to spend money to buy content for a full day.	**Low,** the system is credit-based scheme to provide more opportunity for fraud.	**Medium+,** the system prevents double spending by transferring touchstones between vendors.
Anonymity	**Low,** the system records identities, exchanged amount and time of a transaction.	**Medium,** Broker knows who and where but not what. Vendors know what but not who.	**Low,** Customer's anonymity is not supported.	**Low,** Broker knows who and where but not what vendors know what and who.	**Medium+,** C's anonymity is protected from vendor.
Multi currency	**No,** only one currency $.	**No,** must be match with scrip.	**Yes,** converts	**Yes,** converts	**Yes,** converts
Divisibility	**Very High**	**High**	**Very High**	**High**	**High**
Performance	**Very Low**	**Medium**	**High**	**Medium**	**Very High**
Robustness	**Low**, on-line payments	**Low**, on-line payments	**High**, off-line payments	**High**, off-line payments	**High**, off-line payments

Table 1. Comparison of E-commerce payment methods.

Summary

There is a growing need for an effective, efficient micro-payment technology for high-volume, low-value E-commerce products and services. Current macro-payment approaches do not scale to such a domain. Most existing micro-payment technologies proposed or prototyped to date suffer from problems with security, lack of anonymity and performance. We are currently implementing our NetPay micro-payment model and validating this with on-line information vending applications (including E-newspapers, E-music and informational content sites).

References

[1] Belenson, M., Garay, J., Hauser, R., Herzberg, A., Krawczyk, H., et al. "iKP-A Family of Secure electronic Payment protocols". The First USENIX Workshop on Electronic Commerce, New York, NY, 1995.
[2] Blankenhorn, D. Charging for Content, E-commerce Times, www.ecommercetimes.com/perl/story/306.html.
[3] Boly, J. P., Bosselaers, A., Cramer, R. et al. "The ESPRIT Project CAFÉ, High Security Digital Payment Systems". Third European Symposium on Research in Computer Security, LNCS 875, Springer-Verlag, Berlin 1994, p. 217-230
[4] Cox, B., Tygar, J. D. and Sirbu, M. "NetBill Security and Transaction Protocol". The First USENIX Workshop on Electronic Commerce, New York, 1995.
[5] Chaum, D. "DigiCash". www.digicash.com/ 1995.
[6] Dai, X. and Lo, B. "NetPay – An Efficient Protocol for Micropayments on the WWW". Fifth Australian World Wide Web Conference, Australia, 1999. ausweb.scu.edu.au/papers/#technical
[7] Ediberidze, A., Nikolashvili, M., Abuashvili, N. Design of electronic payment systems for using into Internet. EUROMEDIA '99, pp.247-9.
[8] Herzberg, A. and Yochai, H. "Mini-pay: Charging per Click on the Web", 1996. www.ibm.net.il/ibm_il/int-lab/mpay
[9] Hwang, M-S., Lin, I-C., Li, L-H., A simple micro-payment scheme, Journal of Systems & Software, vol.55, no.3, Jan. 2001, Elsevier, pp.221-9.
[10] Kirkby, P., Business models and system architectures for future QoS guaranteed Internet services, IEE Colloquium on Charging for ATM, IEE 1997.
[11] Manasse, M. "The Millicent Protocols for Electronic Commerce". First USENIX Workshop on Electronic Commerce, New York, 1995.
[12] Neumann, B. & Medvinsky, G. "Requirements for Network Payment: the NetCheque Perspective" Proceedings of IEEE Compcon'95, San Fransisco, March 1995.
[13] Rivest, R. and Shamri, A. "PayWord and MicroMint: Two Simple Micropayment Schemes". Proceedings of RSA "96 Conference,1996, theory.lcs.mit.edu/~rivest/
[14] Visa&MasterCard "Secure Electronic Transactions in Visa and MasterCard", 1996. www.bofa.com/spare_change/
[15] www.cybercash.com/cybercash/wp/