Highly Scalable On-line Payments Via Task Decoupling

David W. Kravitz

CertCo, LLC
8205 Spain NE Suite 201, Albuquerque, NM 87109
kravitzd@certco.com

Abstract. Several digital payment systems have been described which attempt to simulate or extend already existing payment mechanisms so as to make them suitable for electronic commerce. Such mechanisms or instruments include cash or coins (e.g., DigiCash, NetCash), checks (e.g., NetCheque), and credit cards (e.g., CyberCash). The anonymity, off-line, and peer-to-peer aspects of some of these systems can introduce security weaknesses and major scalability problems. One approach to security, as taken by the Millicent architecture, is to only allow very low cost transactions. True security, unlike the approach taken by First Virtual, requires clear delineation of the customer and merchant roles. The goal of this paper is to outline an approach which is inexpensive enough to allow for very low value transactions but secure enough to allow for intermediate value transactions, while providing true customer anonymity with respect to merchants and electronic handling of refund requests. Unlike NetBill and the GC Tech GlobeID system, under the default operation of the system the customer in no way authenticates or identifies itself to the merchant, pseudonymously or otherwise. This is an example of the decoupling of tasks used as a basic design principle: Each system component deals directly with only those aspects in its narrowly defined scope of responsibilities, and within this asynchronous system time-consuming or time-varying issues not directly related to the payment flow, such as actual delivery of digital goods, are handled outside of the basic payment flow. After presenting a high-level comparison of our approach to those of two other instant debit systems, GlobeID and NetBill, we give a more detailed explanation of the design criteria and characteristics exhibited by this new approach to on-line payments.

1 Introduction

The purpose of this paper is to suggest a new approach to on-line payment systems. We demonstrate how *decoupling* the tasks associated with digital payments so that each system component deals directly with only those aspects in its narrowly defined scope of responsibilities, can result in the design of an efficient payment system for digital or hard goods. Tasks are also decoupled within system components (as well as across components), so that refunds, delivery of digital goods, requests for redelivery or retransmission of digital goods, aggregate statements, and transaction notarization are handled separately from the basic payment flow. Effective decoupling results in an asynchronous mode of processing which enables optimal allocation and scheduling of resources. This is not done at the expense of security. Rather, this approach leads to minimization of the cryptographic overhead needed to support correctness conditions associated with on-line payment systems.

We consider the system as having three major components: a *Transaction Processing Subsystem* (TPS) (or Transaction Processor T), a *Customer Transaction Subsystem* (CTS) (or Customer C) and a *Merchant Transaction Subsystem* (MTS) (or Merchant M). A suitably initialized copy of the CTS software is installed on the PC of each customer, and a suitably initialized copy of the MTS software is installed on each merchant server. Customer banks or other means to support funding of customer accounts, and merchant banks, are outside the system boundary. Merchants are known to the system and must undergo some sort of registration and certification process. Customers are known to the system only through customer account information, where a mapping of this information to the actual customer is accomplished through coordination with the customer bank or other funding agency.

In the course of this introduction we will consider two other instant debit systems, namely GlobeID [4] and NetBill [6].

Figure 1 depicts a high-level view of the GlobeID protocol architecture. In response to an expression of interest on the part of the Customer which launches the GlobeID Merchant software, a digitally-signed quote is delivered to the Transaction Processor via the Customer. Our approach, unlike Globe ID: provides customer anonymity vis-à-vis the merchant; directly incorporates a mechanism for encrypted and authenticated delivery of digital goods; eliminates handling and archiving of (non-repudiable) merchant quote information on the part of T; incorporates customer signatures on payment requests. Although GlobeID can suppress customer ID information within the proof-of- payment, the resulting anonymity may actually prove disadvantageous since unlike our system a mechanism for secure delivery of the digital goods is not embedded into the protocol. With regard to the fact that within our system we do not elect to digitally sign the merchant quote information, note that even if all parties sign and transactions are fully archived, and a dispute resolution mechanism exists: quality of digital goods is nebulous and expensive to resolve, while quality of hard goods is impossible to resolve electronically. In the GlobeID system, the payment request consists essentially of the signed quote information provided by the merchant. Three critical ways in which this differs from the situation in our protocol is that in GlobeID the payment request is *always* forwarded to the Transaction Processor, it is verified for merchant authenticity by the Transaction Processor, and the Customer does not append any security element to the payment request but rather does this in response to the challenge from the Transaction Processor if it confirms intention to purchase. In our protocol, the payment request is not issued by the Customer to the Transaction Processor if the intent to purchase is not confirmed. Our system collapses payment processing into a single two-way pass between the Customer and Transaction Processor, as explained below.

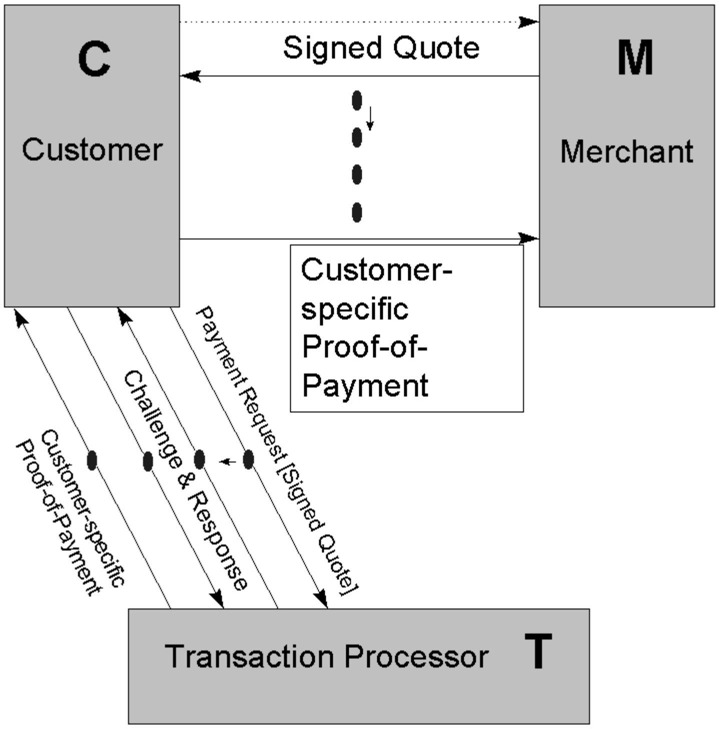

C
Customer
Signed Quote
M
Merchant
Customer-specific Proof-of-Payment
Payment Request [Signed Quote]
Challenge & Response
Customer-specific Proof-of-Payment
Transaction Processor
T

Fig. 1. The GlobeID Protocol

The customer interface with the TPS is through two separate channels: the CTS and TPS correspond via the Internet, while the customer and TPS correspond via telephone through a Voice Response Unit (VRU). This partitioning is advantageous to both security and efficiency. Certain customer support functions are off-loaded to the voice link which would otherwise be conducted through the Internet. These include the changing of PINs, data retrieval, handling of account holds, and reporting of problems. In certain instances, duplication of information through both channels may aid in anomaly detection. The value of the PIN must be conveyed to the CTS for the customer to successfully execute a transaction with the TPS via the Internet. Customer authorization information required for successful access to the VRU should not be held on the customer PC. Initial distribution of this information to the customer should be securely implemented. The customer authorization information must be used to access the VRU for initialization of the PIN (as well as to effect changes of the PIN). This is analogous to having to contact a VRU with personal authentication information in order to activate a credit card. It is important to the efficiency of the system that the interaction between the VRU and the customer is only sporadically required.

The PIN is modified during each (successfully authenticated) transaction for use in the *next* transaction, as an additive function. The cumulative modifier is held on the customer hard drive and is reset to zero each time a new PIN is established through the VRU process. The individual modifier information, one part from each side, is transferred between the Customer and Transaction Processor as an embedded challenge- response process within payment processing. The PIN management process of the Customer, consisting of establishment with the VRU and modification with the Transaction Processor, is depicted in Figure 2. Unlike the standard challenge-response method employed by GlobeID, the per-transaction PIN modifications must be tracked by the Transaction Processor.

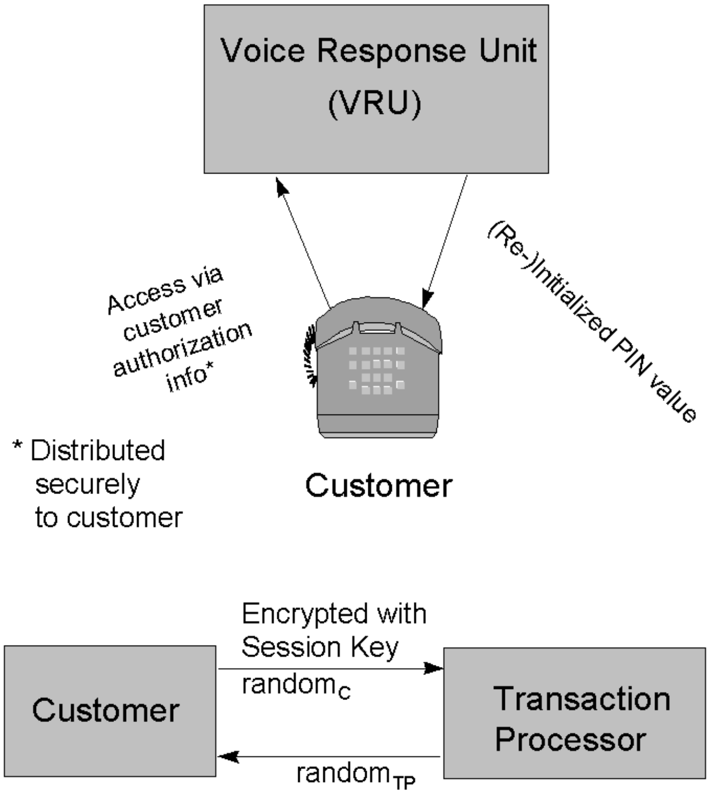

Voice Response Unit (VRU)
Access via customer authorization info*
(Re-)Initialized PIN value
* Distributed securely to customer
Customer
Encrypted with Session Key
random$_C$
Customer
Transaction Processor
random$_{TP}$

Fig. 2. PIN Management

We next briefly consider the NetBill protocol which is an "atomic" approach [6] designed to ensure that a Customer pays if and only if the specified information goods are received intact. An overview of this protocol is presented in Figure 3.

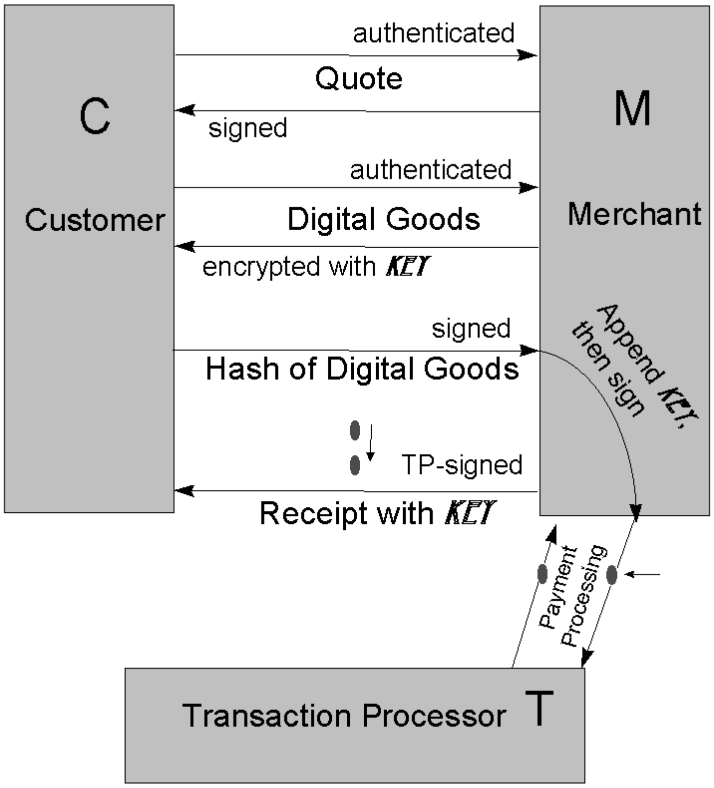
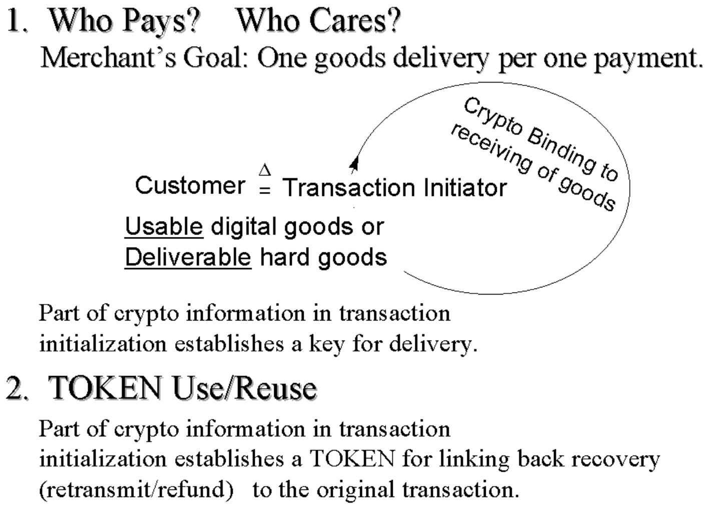


Fig. 3. The NetBill Protocol

The encrypted goods are sent to the Customer up front, with successful payment resulting in a receipt which contains the merchant-specified decryption key, where this receipt is delivered from the Transaction Processor to the Customer via the Merchant. The baseline protocol requires significant overhead as well as a forfeit of customer anonymity. Extensively applied encryption prevents, for example, a passive eavesdropper from ascertaining the matched encrypted goods and decryption key. We will see that in our approach the encryption key is jointly established between the Customer and Merchant, thus obviating the need for (secure) delivery of the key. A key of this nature can also be used to handle authenticated and encrypted communication of the Customer shipping address information which enables the secure delivery of hard goods, which is not within the scope of the NetBill approach. Within NetBill, digital signatures applied by the Customer using a long-term key supply the Merchant with sufficient information under the encryption layer to link together transactions as emanating from the same Customer. In our system, the information digitally signed by the Customer does not pass through the Merchant. Based on the premise that it is cheaper to revoke anonymity to whatever extent desired or required (such as loyalty or frequent-buyer programs) than to provide pseudonymizer/anonymizer services, our system deploys customer anonymity with respect to the merchant within the baseline architecture. Because NetBill does not use PIN-based transaction security, the permanent local storage of the Customer private signing keys poses a more significant security risk than it does in our system. Consequently, NetBill proposes a Key Repository which handles the Customers' private keys by encrypting them in a symmetric key derived from Customer passphrases. In our system, digital or hard goods are delivered in response to payment. In NetBill the Customer can abort the transaction after receiving the encrypted goods but prior to committing to payment. If export restrictions require encryption keys to be of relatively small size, this may introduce a security weakness.

Figure 4 illustrates our basic design philosophy of cryptographically binding the transaction initialization process between the Customer and the Merchant to the receiving of the goods by that Customer in usable form, i.e., digital goods can be decrypted by the Customer and/or the shipping address for hard goods delivery can be cryptographically secured using the same key. The transaction initialization process also provides a cryptographic TOKEN for future referral back to the original transaction.



1. Who Pays? Who Cares?

Merchant's Goal: One goods delivery per one payment.

Crypto Binding to receiving of goods
Customer $\triangleq$ Transaction Initiator
Usable digital goods or
Deliverable hard goods

Part of crypto information in transaction initialization establishes a key for delivery.

2. TOKEN Use/Reuse

Part of crypto information in transaction initialization establishes a TOKEN for linking back recovery (retransmit/refund) to the original transaction.

Fig. 4. Anonymous Process Binding

Note that this paper explains the approach and the design rationale, but does not get into explicit cryptographic design details.

For a survey of previous work in this area of "electronic money," see [9].

2 An Overview of the Communications Flow

The purpose of this section is to give an indication of the communications flow between the TPS and CTS and between the CTS and MTS. The TPS and MTS do not *directly* communicate as part of the basic payment transaction flow. Neither does the customer use the Voice Response Unit channel on a per-transaction basis.

Figure 5 depicts a high-level view of our protocol architecture.

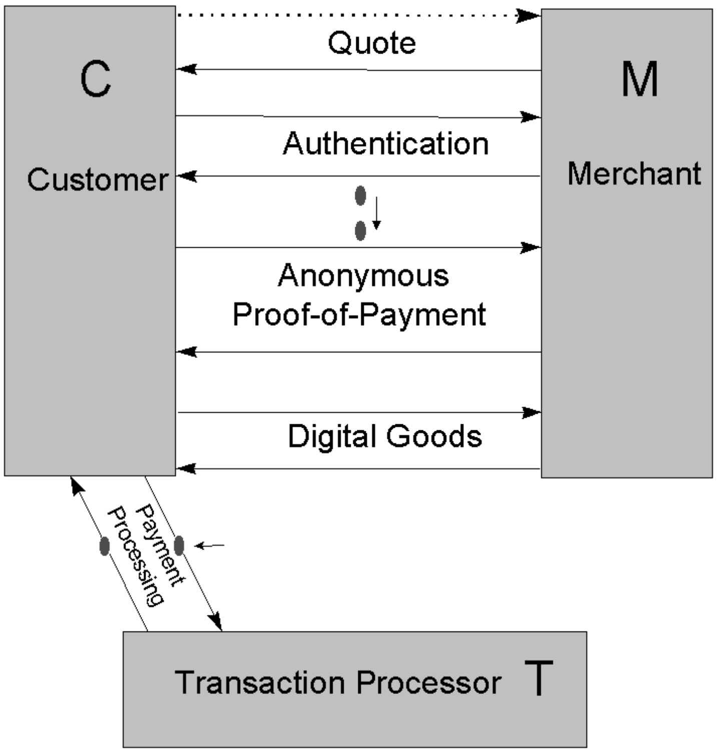


Fig. 5. Protocol Architecture

Below is a general outline of the basic transaction flow, as illustrated in Figures 6,7, and 8:

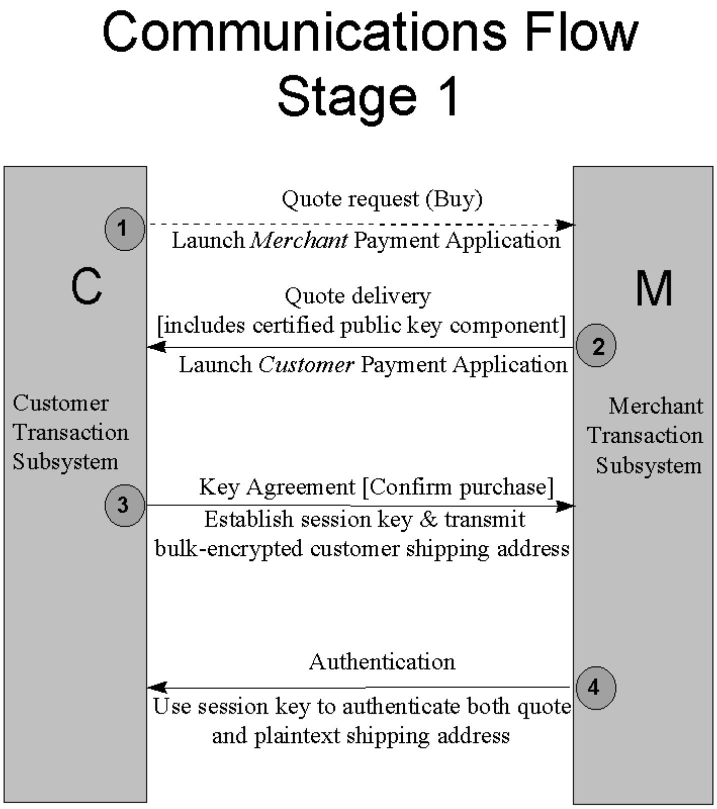
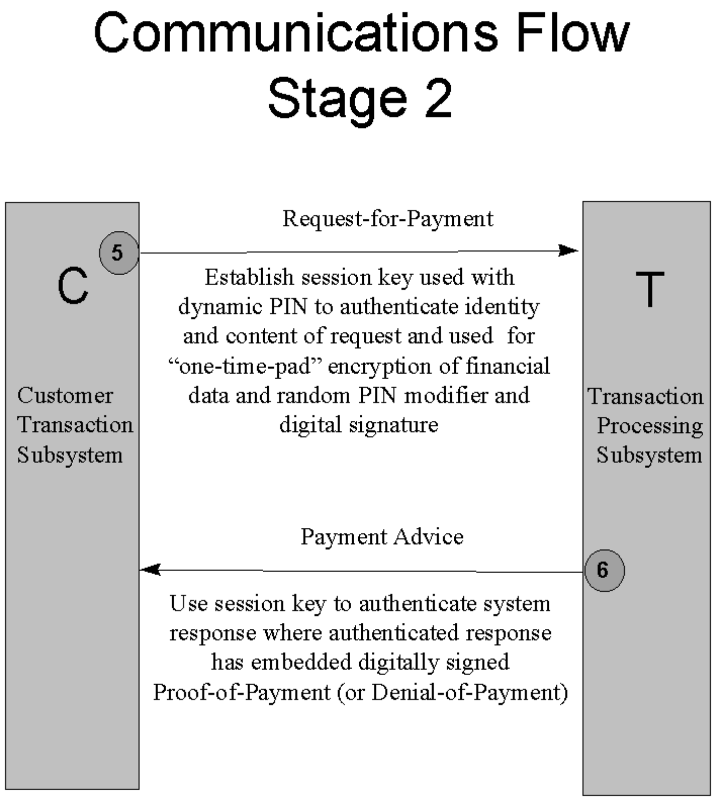

Communications Flow
Stage 1
C
Customer
Transaction
Subsystem
M
Merchant
Transaction
Subsystem
1
Quote request (Buy)
Launch *Merchant* Payment Application
2
Quote delivery
[includes certified public key component]
Launch *Customer* Payment Application
3
Key Agreement [Confirm purchase]
Establish session key & transmit
bulk-encrypted customer shipping address
4
Authentication
Use session key to authenticate both quote
and plaintext shipping address

Fig. 6. Customer ↔ Merchant Session Initialization

1. CTS → MTS: A *quote* or invoice is requested from the merchant via the Internet browser.
2. MTS → CTS: The returned quote includes the merchant certificate and other purchase details. The merchant certificate which binds the public key to the merchant identity is checked for validity at the CTS. If the certificate verifies, the quote information is displayed to the customer.
3. CTS → MTS: If the customer wants to confirm the purchase, the CTS establishes a session key with the MTS by using the certified public key of the MTS. For hard or physical goods the customer shipping address is sent encrypted under the session key. For *digital goods* part of the session key will eventually be used to decrypt the goods as encrypted by the MTS. [Part of the session key is reserved to later re-contact the MTS, if necessary, in order to request a refund or retransmission of digital goods. This would be outside the basic payment flow.]



Establish session key used with dynamic PIN to authenticate identity and content of request and used for "one-time-pad" encryption of financial data and random PIN modifier and digital signature

Use session key to authenticate system response where authenticated response has embedded digitally signed Proof-of-Payment (or Denial-of-Payment)

Fig. 7. Payment Processing

4. MTS → CTS: The MTS authenticates the quote and the received customer shipping address by using the session key.
5. CTS → TPS: The CTS establishes a session key with the TPS by using the public key of the TPS. The customer types in the PIN which is combined with a dynamically changing PIN modifier held on the hard drive. The modified PIN is used in conjunction with the session key to authenticate the customer and the content of the *request-for-payment*. The request-for-payment contains the necessary details of the merchant quote. A digital signature can be applied for transaction non-repudiation.
6. TPS → CTS: The TPS uses the session key to recover the customer-specific information. The database record corresponding to this account is used to retrieve the current modified PIN and other values to test the validity of the request-for-payment. The session key is used to authenticate the response back to the CTS, which includes a digitally-signed *proof-of-payment* (or digitally-signed *denial-of-payment*) to be forwarded to the MTS.

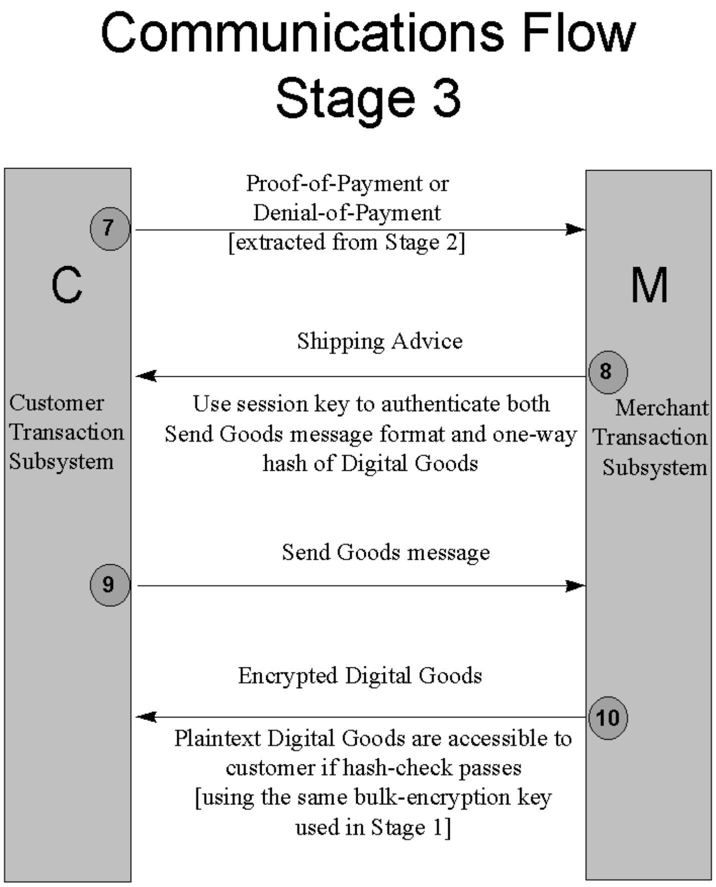

Communications Flow
Stage 3
C
M
Customer Transaction Subsystem
Merchant Transaction Subsystem
7
Proof-of-Payment or Denial-of-Payment [extracted from Stage 2]
8
Shipping Advice
Use session key to authenticate both Send Goods message format and one-way hash of Digital Goods
9
Send Goods message
10
Encrypted Digital Goods
Plaintext Digital Goods are accessible to customer if hash-check passes [using the same bulk-encryption key used in Stage 1]

Fig. 8. Customer ↔ Merchant Transaction Processing

7. CTS → MTS: After using the session key to verify the message from the TPS, the digitally-signed proof-of-payment portion of the message is forwarded to the MTS.
8. MTS → CTS: After verifying the TPS digital signature on the proof-of-payment, the MTS formats a *shipping advice* message which specifies how the CTS should format the *send goods* message if digital goods are indicated. The shipping advice message also includes a cryptographic message digest of the digital goods to be sent. The entire shipping advice message is authenticated using the session key set up between the MTS and that CTS which initiated the quote negotiation transaction. The shipping advice message may also contain information regarding ensuing hard goods delivery.
9. CTS → MTS: After verifying the shipping advice message by using the ses-

sion key previously established with the MTS, the CTS formats the send goods message.

The following is technically not part of the basic payment transaction flow:

10. MTS → CTS: The MTS encrypts the digital goods using the session key set up between the MTS and that CTS which initiated the quote negotiation transaction.

Upon receipt of the encrypted digital goods, the CTS uses the session key previously established with the MTS to decrypt the goods. The goods are checked against the message digest contained within the shipping advice message.

3 Communications Between the CTS and MTS

For the purpose of this discussion we assume that each customer and merchant has a fully functional copy of the CTS or MTS, respectively, with an associated activated account. A significant distinction must be made between the CTS and MTS, however: The CTS has no need to authenticate itself to the MTS, while the MTS needs to authenticate itself and the validity of the quote as coming from that particular merchant to the customer via the CTS. One responsibility of the TPS is to arrange for review of documentation associated with a prospective merchant's credentials before issuing a "certificate" to the merchant which binds the merchant or company identity, as proven in the merchant's credentials, to a unique public key. This public key can then be used by the CTS to assess whether the quote or invoice information actually originated at the MTS.

Considering the shopping process from the inception of a potential transaction, the customer first uses an Internet browser to view available merchandise. After selecting the items of possible interest from a given merchant, marking any preference choices, the customer clicks to get a quote. Because both the customer and merchant are payment system- enabled, the quote should be formatted by the MTS-equipped merchant server and operated on by the CTS in accordance with the system rules.

A design decision needs to be made on how to cryptographically handle the quote negotiation process between the CTS and MTS. If we assume it is solely the CTS's responsibility to verify the authenticity of merchant quotes, the overhead of computing digital signatures applied to quotes on the MTS end and verifying these signatures on the CTS end is unduly burdensome. Analogously to cursive signatures, digital signatures provide verification transferability in two senses: *i) Non-repudiation* - a disinterested third party can reliably adjudicate disputes between the alleged signing and receiving parties; *ii) Relay authentication* - a signed document ultimately intended for a party other than the immediate recipient can be delivered to the intermediary without risk of undetected data-tampering. Until the legal and security infrastructure exists in a widespread manner which is efficient enough to cost-effectively resolve disputes over low- to medium- value transactions, it is unnecessary to support this functionality by incorporating digital signatures into such transactions between customers and merchants as end parties. While it is true that the TPS could play an active role in establishing an internal infrastructure by agreeing to process only those payments which correspond to digitally signed quotes which the TPS successfully verifies and by archiving all quote-related data, in an attempt to counteract the lack of truly effective security of the currently prevalent environment of the customer PC, this would have a significantly adverse effect on scalability of the TPS. Thus it is logical to decouple the TPS's role in processing payments from the responsibility of assessing and maintaining quote integrity. The CTS's role of regulating its response in accordance with the outcome of authenticity checks on merchant quotes can be decoupled from processing or handling of externally presentable proof of such authenticity.

Note that the CTS does not apply digital signatures to any of its communications to the MTS. The model proposed here does not require supporting this additional overhead. Furthermore, the degradation of customer anonymity which results from the merchant being able to link together transactions signed using the same key, is an undesirable consequence of customer-to-merchant digital signatures. The protocols used within the proposed system preserve customer anonymity with respect to the merchant, even if the merchant can completely reverse- engineer the operational MTS. In particular, there is no reliance on an encryption layer applied to CTS - MTS communications in order to guard customer anonymity.

While it is natural to try to emulate existing physically-oriented payment paradigms in the pursuit of constructing digital payment mechanisms, this is not necessarily advisable if one of the goals is to remove the impediments to efficiency engendered by the physical process. Directly translating the physical process to the Internet can also lead to new complexities because of the inherent openness of electronic communications over the Internet. Electronic communications do not exhibit the face-to-face aspects of physical communications. The face-to- face aspect makes the transfer of otherwise anonymous currency more acceptable from the point of view of containing criminal activity. Another physical barrier associated with transfer of tangible currency is the difficulty of perfect reproduction in order to enable double-spending. An electronic off-line approach such as DigiCash [2] which aims for total anonymity of users which abide by the rules, even with respect to financial institutions, invites serious scalability problems when trying to effectively address double-spending issues. Several other systems also exhibit inefficiencies related to trying to faithfully translate existing payment paradigms to the Internet.

An alternative approach to the above is to preserve only those attributes of a physical payment process which are meaningful in an electronic context. One such process involves the commonplace practice of purchasing and obtaining goods which are warehoused and distributed to the customer at a site physically removed from the site at which the payment transaction is conducted. Upon payment, the customer is given a pickup slip to be forfeited at the site of merchandise distribution in exchange for receipt of the goods. In Internet commerce, it is advantageous to decouple or disassociate possession of a pickup slip, i.e., *proof-of-payment*, from proof of identity of the transaction initiator and from proof of identity of the paying party. The real concern or requirement on behalf of the merchant is that the goods are delivered in usable form to a single entity or customer "in exchange for" a single payment: If the transaction initiator makes appropriate provision for payment, the merchant should make a reasonable attempt to complete successful delivery of the usable goods to the transaction initiator. This requirement does not preclude unencrypted communication of the proof of payment or complete anonymity of the customer with respect to the merchant. In this model, the customer is defined to be the initiator of the transaction with the merchant, which may be distinct from the entity making payment. Only the transaction initiator will directly possess the cryptographic key required for decryption of the digital goods, or will get to choose the destination of the physical goods. This completes the transaction loop without trying to tie or couple together intermediate aspects of the transaction.

Trying to tie together the intermediate stages of quote handling, request for payment, and presentation of proof-of-payment processes for secure traversal over the Internet would require additional cryptographic overhead. Another negative characteristic associated with such linkage would be the reduction in the expected rate of successfully completed transactions. In the proposed model the merchant has no need to in any way identify the transaction initiator in order to complete the transaction. Encrypted delivery of the digital goods is provided for, including the capability of retransmission under refreshed keys. However, such delivery is purposefully separated from the core transaction by requiring receipt of a "send goods message" on the part of the MTS.

The TPS's responsibility to prepare *proofs-of-payment* in response to *requests-for-payment* includes taking precautions to ensure that funds are not transferred out of accounts to pay merchants without proper authorization of the account holders. Proper sequencing of requests for payments from a given account is enforced by the TPS. On the other hand, the TPS's sphere of responsibilities does not include checking that the payment requester has the capability to receive the merchant goods by virtue of having initiated the transaction with the merchant. As far as the TPS is concerned, the payment request may be made by an entity which has not communicated at all with the merchant or which has altered the quote-related information given to it by the merchant or which has merely picked off from the Internet quote information requested by the actual transaction-initiating party. The TPS's proof-of-payment generation process does not ensure proper receipt of the goods by a bearer of the proof of payment, nor does the TPS protect this proof of payment against unauthorized duplication.

It is the merchant's responsibility to determine the appropriateness of a received proof of payment. This should include checking that the proof of payment corresponds to a quote actually issued by the merchant. In order to make this check, the proof of payment should include an unambiguous representation of the merchant quote in the form of a computationally one-to-one function of the merchant quote: It should be computationally infeasible to structure multiple distinct quotes which map to the same functional value, or if given a functional value of a quote to find a distinct quote which maps to the same functional value. Application of a cryptographic message digest function to the merchant quote is one way to handle this. Furthermore, a digital signature applied to the proof- of-payment by the TPS endows the proof-of-payment with relay authentication, where in addition, receipt of an exact repeat by the MTS of a signed proof-of-payment does not cause a problem because transactions are indexed by merchant transaction numbers uniquely assigned by the MTS, as well as by the merchant ID.

In line with task decoupling, the CTS need not verify the TPS signature on the proof-of-payment before forwarding it along to the merchant: Since it is the merchant's responsibility to ensure forthcoming payment before executing goods delivery, and not the customer's, the CTS can defer to the MTS the verification of the TPS signature, as long as the CTS can reliably verify the authenticity of the payment status information necessary for the CTS to track. The authentication of this payment status data before transmittal by the TPS and the verification of this data upon receipt at the CTS entails no significant new cryptographic overhead at the CTS or TPS over that already involved in the anonymity-preserving fraud-resistant secure transmission of the payment authorization material from the CTS to the TPS. The CTS cannot depend on the MTS as the sole source of information regarding the outcome of the transaction between the CTS and the TPS, since the MTS' s involvement is not that of a disinterested observer. Furthermore, there is TPS-authenticated information of interest to the CTS such as PIN status which is not transferred in any form to the MTS by the CTS because it has no bearing on the proper functioning of the MTS.

While the MTS should take care in not issuing multiple quotes with the same transaction number, this does not ensure that the MTS will not receive multiple distinct proofs of payment with a shared merchant transaction number, or that multiple requests for payment will not be put through to the TPS by either an individual or multiple parties whether or not each of these results in issuance by the TPS of a proof of payment and a transfer of funds, or that the merchant will be apprised of the status of requests for payment within a timely manner. Unlike the MTS's role of distinguishing and reacting to the specific attributes of the transaction associated with a proof-of-payment, differential handling by the TPS of transactions within an "equivalence class" of transactions which share the same merchant ID and merchant transaction number is not only not necessary but can actually cause significant insecurity within the system. As an example of this it might seem logical to have the TPS, as part of its end of cycle database management processing, scan for repeats of merchant ID and merchant transaction number pairs within its proof-of- payment records in order to issue refunds to accounts which have been debited to pay for requests for payment submitted with duplicate merchant ID and merchant transaction number pairs.

This differential treatment by the TPS of the first versus all latter payments with the same merchant ID and merchant transaction number would introduce a serious security flaw. Consider the potential for misuse of the CTS by prefacing the legitimate request for payment by one which uses the same merchant ID and merchant transaction number but with a smaller payment amount, where the proof-of-payment corresponding to the fraudulent request for payment is suppressed by the CTS and not forwarded to the merchant. This could result in successful receipt of digital goods by the CTS prior to the MTS being notified that the proof-of-payment corresponding to the higher-valued legitimate request for payment has been reversed by being refunded, by virtue of its bearing a duplicate merchant ID and merchant transaction number pair. If all refunds require a digital signature on the part of the MTS corresponding to the merchant indicated within the proof- of-payment as paid, and if all previously non-refunded payments corresponding to the particular merchant ID and merchant transaction number pair are refunded through the TPS, the potential for the above attack can be avoided. For efficiency purposes, all interaction by the TPS with merchants via the MTS is conducted on a batch processing basis.

If the proof-of-payment as received by the MTS is in accordance with the MTS's database with respect to the database entry for the embedded merchant transaction number, then the MTS transmits an authenticated *shipping advice message* which includes information relevant to accessing digital goods via a *send goods message* originating at the CTS and transmitted to the MTS. The shipping advice message may also include the output of a cryptographic message digest algorithm in order to allow testing the subsequently received goods for veracity. In order to more tightly control delivery of digital goods in a form usable to the transaction initiator, the MTS can expire the digital goods encryption key, and refuse to retransmit until the CTS and MTS have renegotiated a new key. Successful renegotiation of this key is followed by a cryptographic acknowledgment to this effect sent from the MTS to the CTS. Successful receipt of this cryptographic acknowledgment by the CTS triggers that it is OK to transmit the send goods message in preparation for receipt of the digital goods as retransmitted by the MTS, since the CTS and MTS have reestablished cryptographic synchronization even if the previous digital goods encryption keys have been cleared from the CTS and/or the MTS. If a refund has been put through by the MTS to the TPS, this may disable the digital goods retransmission capability of the MTS with respect to the given merchant transaction number since in effect the goods have not been properly paid for once the transaction is reversed through refunding.

We have discussed the fact that presentment of a proof-of-payment by a CTS to a merchant via its MTS does not imply that payment is from the account of the presenter; nor does it imply any association to the transaction initiator. In particular, the proof-of-payment travels in cleartext form over the Internet. Consequently, the proof-of-payment does not serve as a receipt in the usual sense. A recipient of a shipping advice message by a CTS as sent by an MTS also has no provable claim of association with the transaction, since delivery of the shipping advice message is also not controlled through encryption or other means. The integrity of the payment transaction is preserved through the transaction initiation protocol. The session key exchanged between the customer and merchant as part of the transaction initiation protocol serves several purposes in order to ease encryption/decryption and tracking requirements. It is used to authenticate the merchant quote, and to authenticate proper receipt by the merchant of the customer's shipping address in the event hard goods are to be sent, where this shipping address information is sent encrypted from the CTS to the MTS. The session key is used to authenticate the shipping advice message and in particular the digital goods content. It is also used to encrypt and subsequently decrypt the digital goods. The session key is also utilized by the CTS in the event that it reestablishes with the MTS a new encryption key for retransmission of the digital goods, or in the event that it requests that a refund for the transaction be issued by the merchant via the MTS to the TPS.

Customer anonymity is not sacrificed, since the merchant via its MTS merely views the original session key of the transaction as a token which allows the CTS to subsequently refer back to the original transaction when addressing digital goods retransmission or refund issues. More specifically, a segment of the original session key is encrypted using a new session key each time, where the new session key is used to authenticate the request for retransmission or the refund request from the CTS, and to verify the returned cryptographic acknowledgment computed and sent by the MTS. A segment of the new session key also serves as the new digital goods encryption key for digital goods retransmission purposes.

4 Communications Between the CTS and TPS

As stated in the introduction regarding PIN changes, reinitialization values of the client (or customer) PIN are conveyed to the customer via the Voice Response Unit associated with the TPS. In between these randomly generated PIN reinitializations, the PIN is modified by an additive function of all previous successfully authenticated transactions since reinitialization. The most current value of this function is held on the client hard drive.

The per-transaction modifier components of this function are randomly generated by the CTS and TPS, respectively. The CTS component is encrypted using a session key which only the TPS can regenerate. It cannot be constructed by an adversary so as to offset the TPS component, since the TPS component is released in response to the request-for-payment which contains the encrypted CTS component. The additive function is reset to a constant each time a new VRU-attained PIN is used.

The TPS automatically resynchronizes the transaction count of the client's account each time an authenticated proof-of-payment (or denial- of-payment) sent in response to the client's request-for-payment correctly verifies at the CTS. The CTS acts sequentially, transaction after transaction.

Repeatedly typing in the PIN incorrectly may result in the client having to call the VRU in order to reinitialize the PIN. Fraudulent use of an account on a PC distinct from that of the target client, may result in the legitimate client having to manually reenter the transaction count to accommodate the skipped transactions. This is good security practice, in the sense that the presence of fraudulent activity should trigger a requirement for special action in order to resume operation. This forces recognition of a breach of normal flow.

The authentication of the client's identity as well as of the origin and integrity of the data within the request-for-payment, is accomplished through use of a computationally one-to-one one-way function of the dynamically modified PIN value and the transaction data. A cryptographic message digest or one-way hash function may be employed for this purpose. The securing of this function is accomplished by using a session key which is randomized on only the CTS end. The session key is generated by the CTS using the fixed public key of the TPS. The session key is used to conceal the client account, CTS digital signature and PIN-related information. The session key also provides the authentication channel back from the TPS to the CTS.

As an aid in minimizing response time at the CTS, the system has been designed to permit similar computationally intensive processing elements to be performed by the TPS in parallel. This is principally due to two factors which relate to the nature of the incoming and outgoing data, respectively:

1. The incoming request-for-payment data is partitioned into plaintext and ciphertext, where the customer-specific data is in ciphertext and the merchant-related data is in plaintext. In order for the TPS to decrypt the ciphertext, it regenerates the session key using the received value of the random public component from the CTS and its securely stored value of its fixed private component. Once this session key is computed, the actual decryption to recover the customer-specific information, the validity-testing of the request for payment using customer data retrieved from the TPS database, and the preparation and authentication of the transaction information needed by the CTS, can all be done very quickly. The transaction information needed by the CTS includes the proof- of-payment in digitally signed form to be forwarded from the CTS to the MTS, as well as information which is of no interest to the merchant or MTS such as reason for non-payment in the event the proof-of payment is actually a denial-of- payment;
2. The plaintext data alone suffices for the TPS to prepare the proof-of- payment (or denial-of-payment) information to be forwarded by the CTS to the MTS. In fact the two potential versions of this message in digitally signed form, one which indicates the merchant is to be paid, and one which indicates the merchant is not to be paid, can be prepared simultaneously as well.

The application of the CTS digital signature to the request for payment is intended to address non-repudiation. The PIN mechanism addresses authorization of the transaction. Once the session key has been regenerated by the TPS, verification of proper usage of the PIN requires very little computation. Verification by the TPS of the CTS digital signature is not necessarily done on a live transactional basis.

The partitioning of the data and the high-level processing strategies are pictorially represented in Figure 9.

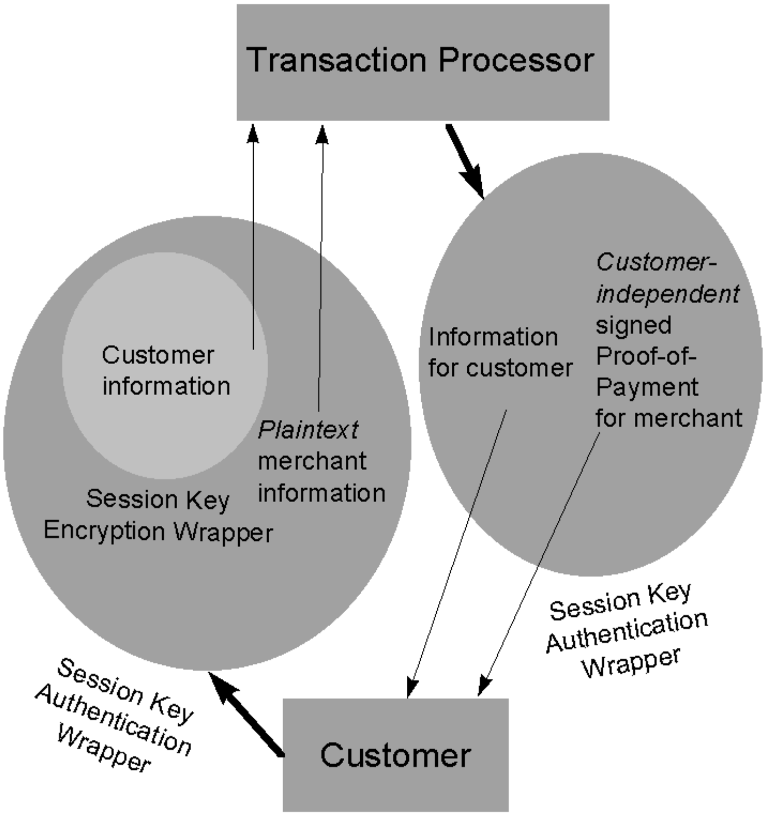

Transaction Processor
Customer information
Plaintext merchant information
Session Key Encryption Wrapper
Information for customer
Customer-independent signed Proof-of-Payment for merchant
Session Key Authentication Wrapper
Session Key Authentication Wrapper
Customer

Fig. 9. Payment Processing Architecture

5 Conclusions

We have proposed a new approach to on-line payment systems which does not attempt to emulate "physical world" transactions. Rather, the approach departs in two radical ways from existing paradigms: First, an attempt has been made to preserve only those attributes of the physical payment process on which the approach is modeled which are meaningful in the context of conducting commerce over the Internet. The second departure from common design practice was to decouple the tasks associated with digital payments so that each system component deals directly with only those aspects in its narrowly defined scope of responsibilities, where the basic payment flow has been streamlined down to include only critical components. Since scalability is an essential feature of effective widespread Internet commerce, the central goal is to enable optimal allocation and scheduling of resources by carefully designing the cryptographic backbone so as to minimize its negative impact on system performance. We used cryptography for authentication of quotes, dealing with shipping of goods, and securing the payment process. This minimal cryptography is sufficient to achieve non-atomic binding of two types, namely, to receiving of goods and to recovery transactions.

References

[1] CyberCash, URL: http://www.cybercash.com/

[2] DigiCash, URL: http://www.digicash.com/

[3] First Virtual, URL: http://www.fv.com/

[4] GC Tech GlobeID, URL: http://www.gctec.com/us/Technical/: FAQ about the GlobeID Technology; URL: http://www5conf.inria.fr/fich_html/papers/: Paul-Andre Pays, Fabrice de Comarmond, "An Intermediation and Payment System Technology," Fifth International World Wide Web Conference, May 6-10, Paris, France.

[5] Millicent, URL: http://www.research.digital.com/SRC/millicent/

[6] NetBill, Cox, B., Tygar, J.D., Sirbu, M., "NetBill Security and Transaction Protocol," First USENIX Workshop on Electronic Commerce, July 11-12, 1995; URL: http://www.ini.cmu.edu/netbill/: Sirbu, M., Tygar, J.D., "NetBill: An Internet Commerce System Optimized for Network Delivered Services."

[7] NetCash, Medvinsky G. and Neuman, B.C., NetCash: A Design for Practical Electronic Currency on the Internet, Proceedings of the First ACM Conference on Computer and Communications Security, November 1993.

[8] NetCheque, Medvinsky, G. and Neuman, B.C., Requirements for Network Payment: The NetChequeTM Perspective, Proceedings of IEEE Compcon '95, San Francisco, March 1995.

[9] SPECTRUM, IEEE, February 1997.

This article was processed using the LaTeX macro package with LLNCS style