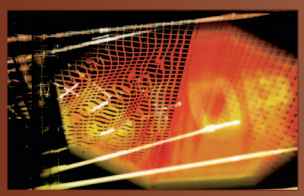
Resolving the Micropayment Problem

Mahesh Tripunitara and Tom Messerges, Motorola Labs



Cryptography-based approaches present a better long-term solution to the micropayment problem than the alternative of restructuring transaction fees.

Transaction fees for micropayments threaten the booming online and mobile economy. Changing how the e-commerce industry imposes online transaction fees can provide a quick fix to the micropayment problem—which occurs when processing fees on such widely purchased small items as iTunes and parking exceed the merchant's margins. However, that simple business approach will provide only a stopgap measure at best because it fails to address the fact that each transaction carries fixed costs.

Implementing applied-cryptography approaches is decidedly more difficult, but it provides a far superior long-term solution. The approaches range from aggregating payments and using tamper-resistant devices to delegating policy decisions and letting users generate payment tokens.

Micropayments provide a means of transferring small monetary amounts and serve as a convenient alternative to traditional payment arrangements. According to the Tower Group, by 2009 the market for micropayments will reach $11.5 billion in the US and $40 billion globally, driven in part by transactions via mobile phones (http://towergroup.com/research/news/news.htm?newsId=820).

A payment ecosystem consists of a *user*, a *merchant*, and a *broker*. A user pays a merchant for goods or services. The broker provides the user with a payment instrument and provides the merchant with confidence that the user's payment is acceptable. A broker is *online* if a merchant must contact the broker when he receives a payment, for example, to validate the payment.

The micropayment problem occurs when the processing costs applied to a user's micropayment overwhelm the merchant's transaction margin. For example, the credit-card processing fee on a 99-cent song could be 35 cents, well above the merchant's 20-cent margin.The processing fee arises from the need for an online broker, such as a credit-card issuer that authorizes the transaction and gives the merchant confidence he'll be reimbursed.

Applied-cryptography researchers have viewed the micropayment problem in the context of e-cash. Micropayments, however, are different from conventional e-cash. The processing cost is the overriding concern with micropayments. Consequently, e-cash schemes that aren't geared to micropayments tend to require heavyweight processing and involve rather esoteric cryptography to fulfill stringent design goals.

Cryptography-based solutions to the micropayment problem typically minimize the online broker's role. While the payment ecosystem remains largely the same, cryptography solutions remove some of the processes and policy decisions from the broker and place them in the hands of the user or merchant.

The business approach to the micropayment problem retains the same payment ecosystem and processes, but it calls on the online broker to lower its transaction cost. While this solution appears to be more easily deployable, cryptography-based solutions present a better long-term solution.

THE BUSINESS APPROACH

The business solution preserves the online broker's role, but proposes restructuring of transaction fees. That is, for a transaction of value a, the fee structure may be $(a \times r + c)$, where the multiplier r and the constant term c may be different for micropayments than regular payments. For example, PayPal charges a regular rate of 2.9 percent plus 30 cents and a micropayment rate of 5 percent plus 5 cents.

However, the broker and merchant can't avoid incurring a fixed cost for every processed transaction. Consequently, the solution is ineffective for micropayments with fixed costs that overwhelm the margins. In the case that $c > 0$, the constant term in the example transaction-fee structure makes this fixed cost explicit. Even if a broker sets $c = 0$, the fixed cost can't be avoided.

The business approach also presents fairness issues. Consider the case of PayPal. If a merchant sells two items, one for $15 and another for $5, the cheapest transaction fee he can receive is $1.04 (using the regular rate for the $15 and the micropayment rate for the $5). However, if another merchant

sells two items, each for $10, the cheapest attainable transaction fee is $1.10 (using the micropayment rate for both items). Even though both merchants had two transactions totaling $20, the second merchant incurred 6 cents more in transaction costs, leading to a perception of unfairness.

CRYPTOGRAPHY-BASED SOLUTIONS

In contrast to the business approach, cryptography-based solutions minimize the online broker's role during payment transactions using a combination of several solutions.

Aggregation

Some cryptography-based solutions let the merchant aggregate several user payments into a single payment. Thus, when the merchant seeks reimbursement from the broker, he presents only an aggregated token, thereby lowering his transaction costs. Figure 1 shows an approach to deterministic aggregation using cryptographic hash chains.

Another approach is *global aggregation,* in which a system aggregates a single user's payments across multiple merchants. An example of this is Peppercoin, in which the user creates lottery tickets such that the winning amount is an aggregated amount, say $10. If he creates 1,000 lottery tickets, then each ticket is probabilistically worth 1 cent.

When the user needs to pay a merchant 1 cent, he sends the merchant a lottery ticket. The broker is involved in creating the tickets, a merchant can instantly recognize a winning ticket, and a user can't forge tickets to be winners or losers. A merchant contacts the broker for reimbursement only when the merchant gets a winning ticket.

Tamper resistance

Tamper resistance also minimizes the online broker's role. Research and development advances in tamper-resistant devices make it possible for a broker to have his agent collocated with the user's device.

In this *observer-representative model*, the observer is the broker's agent, and the representative is the user's agent. Both the observer and the representative are involved in payment transactions, and the observer interfaces only with the representative. This ensures that the observer doesn't compromise the user and the representative doesn't compromise the broker.

An example is for the broker to store payment tokens that he has generated in the observer. When the user needs to make payments, the representative asks the observer for payment tokens. The representative can scrutinize the tokens before sending them to the merchant.

The broker can ensure that if a user compromises an observer's tamper resistance, the damage is limited to that observer and its contents at the time it's compromised. The broker can also detect that particular payment tokens are from a compromised observer when a merchant files for reimbursement.

Delegating policy decisions

Another cryptography-based solution is to have merchants make policy decisions (for example, validate credit cards and digital certificates) that the broker would otherwise make online. A merchant might decide that he will check against the broker's online revocation list to verify that the user's credentials haven't been revoked or compromised only when a user's (aggregated) payment crosses a certain threshold (say $5).

When the merchant checks with the broker, he'll likely incur transaction fees. The merchant is willing to undertake the risk of not being reimbursed up to a threshold that he sets, at which point he trades off the risk for the transaction cost. The merchant can incorporate other considerations in his policy, for example, past losses and the potential for large aggregated losses from several small losses. This is already practiced in the context of credit cards.

User-generated payment tokens

Due to advances in storage and computing power, small devices such as mobile phones and smart cards can generate payment tokens. Users can now count on such devices performing public-key operations in practical situations.

In the case of mobile phones, over-the-air downloading of a payment instrument, public-key certificates, and security software updates is also possible and becoming more cost-effective. In conjunction with tamper-resistant components (collocated with

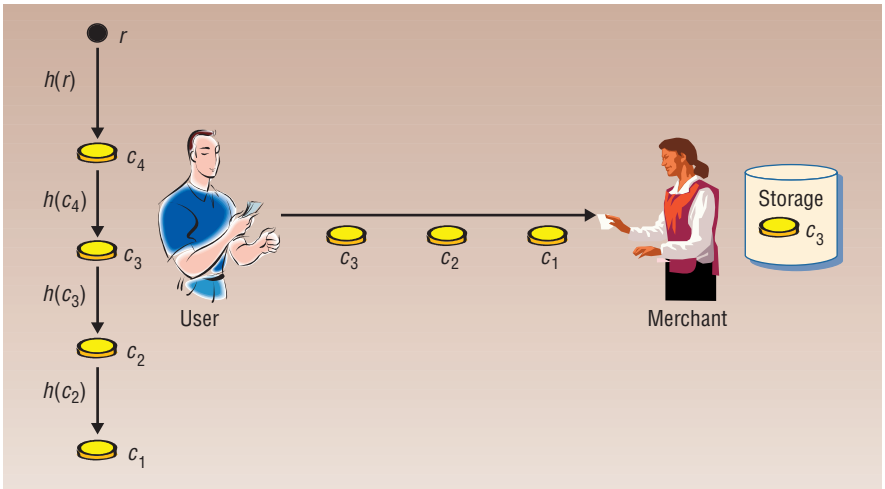


Figure 1. Cryptographic hash chains. The user repeatedly applies h( ) *starting at a random number* r *to generate a hash chain. The entries from the hash chain are used in reverse as payments. The merchant must store only the last payment to prove that he received all previous payments from the hash chain.*

the user's device) that act as the broker's agent, it's possible to meet security requirements while minimizing the tasks that the broker needs to perform.

For example, the broker can store a certificate in the observer that permits the observer to mint payment tokens on the user's behalf. The tokens are cryptographically bound to the certificate through means such as including a hash of the certificate in each token or requiring that each token be associated with a certified signature.

The merchant verifies the certificate and the tokens' integrity to validate the tokens. In conjunction with aggregation and delegation of some policy decisions to the merchant, the approach greatly minimizes the need for the broker to be online, thereby providing significant cost savings to the merchant.

Cryptography-based alternatives are the most appropriate long-term solution to the micropayment problem, especially compared to the business solution of restructuring transaction fees. We acknowledge that the business solution is easier to deploy in the short term. Cryptography-based solutions face tremendous deployment challenges: They need more market testing, they significantly change payment-ecosystem processes, their legal implications are unclear, and they'll require new kinds of devices and processes for merchants. Nonetheless, the long-term payoff of investing in such sound solutions will be tremendous.

Mahesh Tripunitara *is a principal engineer in the Security and Privacy Technology Lab within Motorola Labs. Contact him at tripunit@motorola.com.*

Tom Messerges *is a distinguished member of the technical staff in the Security and Privacy Technology Lab within Motorola Labs. Contact him at tom.messerges@motorola.com.*

Editor: Jack Cole, US Army Research Laboratory's Information Assurance Center, jack.cole@ieee.org; http://msstc.org/cole