SPKI based solution to anonymous payment and transaction authorization

Juho Heikkilä and Markku Laukka

Key words: SPKI, payment, anonymity, transaction, authorization, certificate

Abstract: *Network commerce has been raising its head for a few years and has adopted credit cards as the most common means of payment. However, on an open network like internet, credit card based systems have problems. Especially problematic is the authentication of the parties involved. Privacy issues are important since the customer is always identified with each payment. A few solutions have emerged, but none of them has gained an upper hand and some have withered away as interest towards them has died. In this paper we discuss the limitations of current systems and the criteria for a good electronic payment system. Finally we propose a new, SPKI certificate based payment system that solves most of the problems we found.*

Introduction

The emergence of Internet has brought us on the verge of a totally new era of commerce. Services can already be made available over the network, but the provider would like to be able to charge for them. However, the Internet was not designed as a commercial network and is open and insecure by nature. There is no such thing as a robust global method of secure payment that would guarantee the transfer of funds. Many systems have been proposed but none has gained global popularity. This has lead to the usage of credit cards as the standard way of payment on the Net.

Today, when people talk about electronic commerce, they most often mean ordering something over the network and paying for it with their credit card. This hardly differs from the old-fashioned mail-order commerce, the only difference being that the order is made over network instead of phone or mail.

Taking a look back in the real world, we can see that doing trade used to be quite simple: the customer met with the merchant, chose the product or service he wanted and paid for it with money vowed for by a higher authority, usually the state treasury. This approach had its problems, not all people were law-abiding, and therefore it was not safe to travel with large amounts of money. Also, when trading in large quantities, it was impractical to hand over large amounts of payments. Medieval Italy was among the first places where money transfers between merchants are known to have been made on paper. In 1200 AD a meeting with a banker was required, but by 1600 written authorizations were common. [1]

Eventually, written transactions became a tool for the common man in the form of cheques that are still in use today in many parts of the world. Later came other payment methods like bank and credit cards. What is common to all these replacements of money, is that the merchant needs to take some additional steps to get his money.

Credit cards

The card (be it bank or credit card) is a sort of a proof stating that the person to whom the card

was issued has the right to use some account. Its primary purpose is to allow the customer to pay for services without carrying cash. If cash is stolen, anyone can use it, but if the card gets stolen, it should not be usable by anyone else. Therefore, before the merchant accepts the card as a form of payment, he should check that the required conditions are met [2]. The merchant needs to check that the card is valid, the user is the one to whom the card was issued and that there is money or credit still left on the account.

Since a card is issued to a person i.e. a name, the only way to check for the authority is to check whether the person using the card is the person to whom the card was issued. In on-the-place trade this is usually easily accomplished with some form of proof of identity. The merchant also has to check that the card is valid and that the sum is chargeable. This requires the merchant to contact the issuer of the card. The merchant can choose to skip this, and by doing so, assume the risk of not getting the money, should foul play be involved. Therefore, large purchases are usually checked by calling the issuer or contacting the issuer via telecom networks, where applicable.

As summary, the use of credit card requires a quarantee of the identity of the cardholder and of the validity of the card. Fulfilling these requirements can be problematic even in traditional transactions. It becomes even more difficult when making transactions on an insecure network, where the authentication of the parties is yet harder.

We argue that stronger identification of the customer per se isn't the right way of developing the electronic payments systems. We propose a system that is anonymous, can be connected to the current banking system and makes the payment safer not only to the merchant but also to the customer.

The rest of the paper is organized as follows: In section 2 we discuss problems of electronic transactions in more detail. In 3 section we present some criteria that we find appropriate for a new payment system. Section 4 will take a look at already published solutions in the field. Section 5 will introduce our suggestions for solutions and in section 6 we evaluate the properties of our system in the light of the criteria presented. In section 7 we propose some directions of future research we found especially interesting. Finally, in section 8, we present some conclusions.

Problems

We will discuss four problems with credit card based payment systems. These include merchants excessive rights to the customers account, the huge databases of customers, the insufficient support for micropayments, and the problems many users face with having to handle multiple cards. Finally we argue that the current credit-card-based system requires unnecessary amount of trust from the customer due to the presented problems and can thus be an obstacle on the road of expansion of e-commerce.

Merchant gains excessive account rights

Since no signature or equivalent is currently involved in network trade, there is no way to authorize a single payment, but the customer is forced to give the merchant all the information needed for any transaction. That is, the customer has to give the card's number and expiration date to the merchant. With these numbers the merchant can make the money transfer to get the agreed sum for the services provided. Or make any other transaction he wishes. Therefore, taking into account the nature of Internet, making purchases over the network with a credit card is a risky affair for the customer. Giving credit card information to an unknown merchant may lead to frauds. The information might leak into the hands of an eavesdropper, or the malicious merchant might misuse it, either by multiple charges or never providing service, or both. The customer can do nothing to control the usage of his credit card by the merchant or any other party that gains access to the card information.

Most reputable companies are quite trustworthy and do not abuse customer's card as such, but there is another problem related to the databases. The greatest fear of the credit card companies today is not that a merchant would misuse the information he has gathered, but that someone else would gain access to the database and use the card numbers in a spread fashion that would be hard to detect, not to mention trace.

Databases about customers

When paying at a shop with cash, the customer can keep his identity relatively secret, at least no clear traces of transactions he would not like the public to know about are left behind. When paying by a traditional bank or credit card, the customer's identity is revealed and stored in the merchant's system. The merchant is expected to identify the possessor of the card in order to accept it as payment. This doesn't usually appear as a threat. However, when information is collected in connection with every payment, the databases created offer a considerable amount of information that could possibly be used to control people's lives. Should some malicious party gain access to databases containing such information, combining these could reveal much about the person in question. In most civilized countries this would be considered an invasion of privacy. The merchant is required to keep this information confidential, but are there really any consequences for the merchant failing to safeguard the database? And once the information is out, it is practically impossible to stop it from spreading.

Any database containing information about thousands or even millions of users is sought after and entities are willing to pay to get their hands on it. Since the information has such demand, selling it could be very profitable. There are companies selling databases of customer addresses [3]. Today also collections of email addresses are sold on CDs for mass advertising. Less reputable companies are willing to pay for these in order to find a few new customers. Tomorrow, on the black market one might be able to buy collections of credit card numbers and expiration dates. Consider for example the effects of someone being able to copy and distribute amazon.com's customer database.

These are actually problems in the current credit card system, not problems on network trade. The only new problem in the network trade is the lack of signature. As can be seen, the problems of identification and storing data in databases are already present in the system. We would like to point out that as more systems get online, the risks spread even into the conventional stores. Many stores will have a conventional store and also make products available on the network. The customer data is likely to be stored in a single database. If such databases can be accessed through the network they can be broken into and identities and purchase information can be stolen. Though the risks are slightly different than in making the transaction over the network, using the card in a conventional store does expose the user to some of the same risks and exposes them to some new ones, like a dishonest cashier. Further, secure communication can not protect the parties from attacks that aim at data stored in the merchant's systems. Since the merchant does not really need this information in particular, it would be safer not to store it in the database.

In the section 5 we propose a solution where the right to use an account is proved using public key infrastructure in combination with smart cards. This allows us to introduce a signed payment system that makes it possible for the customer to remain anonymous toward the merchant.

Micropayments

Possibilities of e-commerce not yet utilized are the services that would be available on a pay-per-view basis. On each visit to a web page a small amount of money would be charged from the customer's account. These small payments are also called micropayments [4]. Some commentators are of the opinion there will never really be a need for such payment [5][6]. We, however, consider them as a possibility for developing services on the internet and as such a feature in the payment systems worth supporting. In the search for a flexible payment system, we feel that micropayments should not be completely ignored.

Credit card payment is poorly suited to small payments required to make these kind of services work in a trustworthy fashion. The main problem is the fixed expense per transaction. Furthermore, if such payments were common place, the spreading of one's credit card number would be a considerable risk, making the now fragile system even more so. It is also noteworthy that tracing the thousands illegitimate micropayments is next to impossible.

The unsuitable payment system as such is not the only obstacle on the way of micropayments. There is the question of ease of use: if a small amount of money is to be charged whenever

one does something like plays a sound, views a picture, displays a document etc., how do we balance between the ease of use and the protection of the user from excessive charges. Nobody wants to check the amount and click on OK for every visited object, nor does the customer suddenly want to find out he paid $10 for each of those sounds he listened to.

Problem with multiple credit cards

Current credit cards collections aren't very user-friendly. You can have a large number of cards that are valid in different places and they all have their own unique PIN-numbers. No wonder many people keep those numbers on paper. Nowadays, more functions are added to one card. Even so, current credit cards are quite limited because the functions added are permanent and the functions available on the customer's card are closely bound to the company issuing the card. With current smart card technology the addition and removal of functions would be more dynamic and at the same time the amount of PIN's needed to remember would decrease. The PIN's aren't a problem when cards are used directly to pay things on the internet or in a real life shop.

Trust needed in electronic payment

The problems of excessive merchant rights and databases, as discussed earlier in this section, are closely connected to the issues of trust. Normally, there are three parties involved in an electronic transaction: a customer, a merchant and a bank, where the customer has his/her account. All parties need to trust each other to some extend for a transaction to take place. Since we are not paying by cold hard cash, the merchant needs assurance that he will get his money. The customer needs to know that he is charged no more than the amount he authorized. The bank wants to be certain that money is not lost or duplicated along the way.

We argue that nowadays in the e-commerce the customer needs to trust the merchant more than necessary due to the first two problems mentioned in this section, i.e. the merchants databases of credit card numbers and customer information. After a single payment to an internet-store a customer can do nothing but trust the merchant not to use the obtained information against the customer and also to guard it adequately. Especially in the context where, in many cases, all the information the customer gets from the merchant is through the merchant's web page, it's a real leap of trust to give one's credit card number to such unknown party. Why should people accept this, when it isn't necessary with normal cash based payments either?

In fact, many people are concerned about the idea of giving their credit card numbers to anyone[8]. Especially not to an internet based company of which they have no experience whatsoever. Several studies suggest that customers are worried about the security of e-commerce and would, for example, increase their internet usage if their account was better protected [7][8]. We argue that the development of new means of making transactions have to be designed for e-commerce to really reach the volume for which it has the capacity.

Problems taken together

We argued that there are four central problems in credit card transactions today. They were the following:

- The merchant can potentially make transactions from the customers account without permission
- The merchant gets a huge database of information about the customers
- The support for micropayments on the internet is inadequate
- Too many cards are difficult to handle.

We also stated that because of the deficits of the current system, the practitioners of e-commerce can't utilize all its opportunities. That's due to unnecessary trust required from the customer towards the merchant.

In section 3 we are going to propose a set of criteria for an electronic payment system that could meet the challenges of digital world.

Criteria

There are numerous criteria that a universal payment system should fulfil. We divide the requirements in four categories (though we

admit that many criteria would fit in more than one category)

- Customer based requirements, what the customer expects from the system.
- System requirements list some general requirements, some of which are in fact common to any distributed system.
- Security requirements are there to prevent misuse of the system, these could be considered to be required by the bank or whichever party is responsible for the money.
- Other requirements list some things that did not seem to fit the above categories.

User's needs based requirements

Retaining control – The user wants to be certain that she is in control of the situation. One of the most feared spoofs on the network is that the merchant charges the customer's credit card more than she intended after the merchant has got his hands on her card number and expiration date. The customer shouldn't have to trust the merchant with her credit account. Therefore, the authorized sum must come from the customer, not the merchant, and the system must make a difference between charges so that the merchant can only charge the amount once.

It should, naturally, be possible to make payments that are more complex, like once per month, $50 each, for the next six months. For usability, it might be nice to have some feature keep track of such transactions, but how this should be accomplished is beyond the scope of this paper.

Anonymity – The customer should be able to deal with the merchant without revealing her identity. Remaining anonymous to the bank, as well, is much more difficult, and in general is not as essential. The bank, after all, wants to know from which account the money is to be drawn. Using digital cash would provide even higher anonymity.

Naturally there are occasions where anonymous dealing is impossible. In such cases, the checking of identity should and can be done using methods other than the credit card.

It should be noted that anonymous payments do not render the shops bonus cards useless. These can still be used, the only difference is that the customer retains control over whether he chooses to display it or not. Such cards can also be made anonymous at least to some extent.

Last, we would conclude that in the era of digital networks it would be possible to provide proofs of age independent of proof of identity.

Privacy differs from anonymity thought the two are closely related. When anonymity is used to protect the customer from the parties involved, privacy is protection from parties not involved. Transactions and possibly delivery as well, should be done over secure channels, so that nobody else will see who is purchasing what and for which price. Even the bank does not need to know what it was that the customer paid for.

Proof of transaction – the user wants to be able to prove that the money has indeed left her account and that she should therefore get the service she paid for.

Merchant verification – When making a purchase and paying, the customer would also like to be sure that the merchant really is who she thinks he is. That he really is a merchant and is going to ship the goods, and that there is nobody in the middle stealing the money. Unlike the customer, a honorable merchant has nothing to gain from staying anonymous. We would rather feel that a merchant would gain in trustworthiness by being strongly identified. We claim that the emphasis of a brand is even more important in a network society than it is in a physical one.

Ease of use – The system should not require any training, at least not on the customer side. Usability is the next important thing to trustworthiness when designing a system for general public.

The old argument that usability and security are contradictory things still dwells in people's minds, but we claim that things are not black and white even in this field. We argue that usability should be part of any security system design, especially those involving the end-users. If security is made too difficult, people have a way of social engineering around the difficulties, in this case security.

Acceptability means that the payments made by the system should be widely accepted. Why would anyone like to try a monetary system that

is only valid in few services if he can use a more conservative method he feels is safe and usable. Few people carry dozens of credit cards with them, though a few cards are not uncommon.

Micropayments – One of the expectations of network commerce is the possibility to pay very small amounts of money. The network would be an ideal place to provide services and charge them on the pay-per-view basis, but for people to really use such services, the charge would have to be of the order of pennies. The expense of a credit card transaction today limits the minimum amount of payment to a few dollars. For the "micropayments" to become possible the cost per electronic transaction would have to be negligible. This can be achieved when the transaction is fully automated. The system should be able to count fractions of pennies to make room for exchange rates and charges in currencies other than dollar.

Multiple payment types – There should be more ways to pay than one. Basically since people are used to paying by cash and by credit, they want to keep doing that in the future.

Not all the payment types have to be implemented in the same package, but rather we call for an open standard that would allow for creation of new payment methods that could be used in collaboration with the existing ones.

Currency independence – The Internet is an international arena, and therefore the use of more than one currency should be taken into account when designing a payment system.

System requirements

Efficiency – The system must not pose any noticable delay in the transaction. Since network access times are already quite high, adding another such delay due to payment would be unacceptable, especially in pay-per-view trade where such delays might take most of the time used for browsing.

Availability – The system must be available for use at all times. As we claim that the system should be usable in on-the-place as well as on the network, it must be just as robust and reliable as the conventional card payment systems.

Flexibility & Convertibility – A fair assumption is that, at least in the beginning, there will not be one system but many, and therefore moving money between systems would be something the users probably would like to be able to do. Some people like to carry many cards with them, but some prefer to always pay with the same card. Therefore that card should be valid in most places, if not directly then by conversion of some sort.

Scalability – The system must not require any extensive infrastructure beneath it, making it possible to launch it on small scale first. But as the system should be an international solution, it must be able to grow to a global distributed system without sacrifices on access times.

Transferability – It would be nice if the system allowed the transfer of funds from one party to another without the interaction of a currency server.

Security requirements

No forging – Any monetary system must be protected from frauds. In a digital world making duplicates is easy and often desirable for backup purposes. Money, however should not be duplicable. This means that once the user has spent the money, he should not be able to spend it again, even if he has backups of his purse.

No stealing – The system should guard against the possibility that someone gains access to your account or the money in transfer to the recipient. Or, as more appropriate in the digital world, is unable do do anything with that access.

Tracking enablable – It has been argued that digital money is too anonymous and a system allowing for revocation of anonymity []. In cases of catching criminal activity such revocation would certainly be useful. However, such a revocation must be tightly controlled and require the co-operation of multiple independent parties. It has also been argued that a court order would be required to revoke the anonymity [], which is in general a good idea, but which requires more thought in the international environment.

Other

Checking coverage – It should be possible for the merchant to check the coverage of the payment. In case of digital cash that would be validating the "coins" and in case of a cheque, asking the issuer if the cheque is good. The

system must however be operatable without such checks, in this case the merchant assumes more risk to himself, which is the way the system works today.

Off-line operation – The system should not require an online access to the currency server or banking service.

Open standards – If the system is to gain global acceptance, it might be a good idea to provide the basic system as an open standard, allowing different parties to produce multiple, interoperable implementations. Open and expandable standard would also allow for the development of new payment methods.

Ease of integration – As the world is spun in fibres, programs that cost to use will emerge. It would be plausible to assume that the same payment method would be needed to integrate to such programs. Therefore the interface should be simple, we see this to be in the interest of the banking industry as well.

existing solutions

Published means of electronic payment can be grouped into three broad classes. These are electronic currency systems, credit-debit systems and systems supporting secure presentation of credit card numbers [7].

Electronic currency

Electronic currency systems use currency certificates as electronic coins [1]. These certificates have to be signed by a trusted bank or other such organisation. There is wide variety of solutions, but the majority's basic functions are pretty much the same. We describe here the eCash system as an example and then evaluate the pros and cons of electronic currency systems as a concept.

Ecash is an electronic cash system of the company eCash technologies, Inc.. The system consists of three entities:

- The bank, who mints coins, validates existing coins and exchanges traditional forms of money for eCash.
- The customers who have accounts in the bank. They can transfer money from their traditional accounts and deposit it as electronic coins to their eCash wallet. Now, they can pay with the coins they possess when dealing with a merchant who accepts eCash payments.
- The merchants who accept eCash coins for a payment and can deposit the coins in a traditional account through the bank.

To be able to make a payment with the eCash system, the user must first transfer money from her bank account and save the coins is her electronic wallet located usually on the users hard disk. The minting is performed using the blind signature technique [2]. As a result the coins can't be traced back to the user who withdrew them.

Now, the user has money in her e-wallet and can use it when doing business with a merchant accepting eCash. Before the merchant gives the customer the service, she deposits the coins to the bank she has contract with and the bank checks the validity of the coins i.e. if any of the coins have been used before.

Ecash-system is based on RSA public key cryptography. This means that every user in the system has a private-public-key pair. In all transactions the coins are crypted with the public key of the receiver. This ensures that the coins can't be decrypted by anyone but the authorised party, which makes the system safe from eavesdropping and message tampering. However, it doesn't protect against an attack with a false identity. In such attack the intruder misleads the other party to believe that she is doing business with a trusted party, for example a bank or a known merchant.

The characteristics of the e-currency systems on the conceptual level are mostly linked to the fact that an e-currency coin is valuable in itself. On the positive side is the anonymity of the money. There is no need to demand identification during the transaction. The possession of the coin is sufficient. There are still valid reasons to enable identification such as the issue of money laundry [16]. Moreover, when no identification is needed, the off-line operation presents a bigger risk to the merchant.

Payment performed by transferring electronic coins makes the whole idea of overcharging impossible, and also very small payments (micropayments) are possible. The

double spending is controlled by big databases containing information about every used coin. This database is used if the coin is used more than once. The anonymity of the coin is reversed using the information stored in the database. However the database required can become large in size, which forms a problem in scalability.

The user of an e-currency system really carries money around in his electronic wallet. Usually the wallet is in the users PC. As a result there is also a possibility of destroying or losing money due to system malfunctions or hardware problems [17]. If one accidentally deletes his money folder, nothing can be done to recover coins because of their anonymity. Furthermore, because a coin is valuable as itself, a coin stolen directly from a hard disk is completely valid. Another source of inconvenience is the possibility of leaving your money to a wrong terminal. One could need her electronic money, when working with a laptop, but if the wallet is in the PC, the only alternative is to contact the bank and ask for more coins from the account.

Even though there has been a lot of hype around electronic commerce, the electronic currencies still haven't become common means of payment. One reason for the lack of critical mass may be that the use of the system has appeared too troublesome to the users. In the eCash system you first need to withdraw coins from your own account to your electronic wallet. Only after that can the coins be used with merchants who accept them.

Credit-debit systems

In credit-debit systems customers are registered with their normal bank accounts to a payment server. After the registration has taken place they can authorize charges against those accounts signing the charges with their secret key. One system using this kind on model is NetCheck [9]. The check format in the NetCheck systems contains the following: - the name of the payer, the name of the financial institution where the payer has an account, the payer's account identifier, the name of the payee, and the amount of the check. The NetCheck system uses symmetric key infrastructure with Kerberos system [6] to authenticate the signatures on the checks.

Credit-debit model doesn't directly provide anonymity, but it can be extended to do so [5]. Our solution is very similar to this model.

Securing credit card transactions

Credit card payment is the traditional approach to the payment in the net. SSL has been used to protect credit card numbers from eavesdropping. It has been possible to verify the identity of the merchant by checking her identity certificate, but it hasn't been mandatory.

Lately a new SET specification was announced. It is "a technical standard for safeguarding payment card purchases made over open networks"[4] and is designed to add confidence to the credit card payment process. Merchants and customers have to register to be able to use the system fulfilling SET standards. The standard requires a stronger identification of the parties in the transaction process. It also ensures that the parties taking part in the transaction process are all authorised to do so. However, also with SET the merchant has a possibility to acquire at least some of the customer's credit card info.

The big advantage with credit card based systems is that the customer doesn't need to be registered to a payment service, all they need is a credit card. The traditional, SSL based transactions still have all the limitations discussed in chapter 2. SET is a better alternative when some security issues are concerned, but the obligatory registration decreases its major advantage, the ease of use.

ISSUES IN IMPLEMENTATION

In this chapter we will first present the key techniques enabling our solution and later discuss the parts of solution where multiple paths could have been selected.

Enabling technologies

The two basic building blocks of our system are public key cryptography and authentication certificates.

Public key cryptography

Public key cryptography is based on concept of keypairs containing one public and one private key[14]. The keys complement each other so

that to open a message encrypted with a public key one needs the corresponding private key and vice versa. Therefore, crypting a message with ones private key can be used as a electronic signature. Now, one can make valid electronic contracts.

Authorisation certificates

A certificate is a signed fixed form statement about the properties of some entity [13]. There are two major types of certificates. First, it can be an identity certificate stating the signer's belief that the public key and the identity mentioned in the fields of the certificate are connected. Another forms of certificates are the authorisation certificates. An authorisation certificate delegates some right or property of the issuer to the subject. The certificate is signed with the private key of a party having a right to the resource. There are different standards for a certificate concerning the fields involved. In our solution we are using the SPKI certificates[12]. Where the following fields are included: issuer, subject, authorisation, delegation right and validity.

The certificates are used in our solution in the following way. A bank makes a certificate where the right to use an account is granted to a public key of a certificate holder. By presenting the certificate and proving the possession of the corresponding private key the certificate holder is considered to have a sufficient proof of the right to use the account. Furthermore, the owner of such certificate can delegate the rights defined in the certificate to another public key. She can also delegate a subset of the rights. This kind of delegation can also be used as a payment, as we shall see later.

Issues in implementation

The implementation of the system we propose requires careful consideration of the following issues. First, one needs a secure way of providing the correct public key of the merchant. Second, the certificates must be connected to form a complete chain. Third, one must make sure, that all certificates in the chain are valid. Fourth, we consider the possibilities to use this certificate-based system to perform mictopayments.

Providing correct public keys

The use of authorisation certificates requires the keys of the participants. A payment can only take place if the customer has her private key available to do the signing of the certificate and the public key of the merchant is known. The private key of the customer could be stored into the memory of the customer terminal or into a smart card. The smart card has some benefits compared to the memory of the terminal used. Smart card is a convenient way of using multiple (trusted) terminals safely. It isn't necessary to get the public key outside the card. This makes the securing of the key much easier.

The providing of correct public keys is the bigger challenge of these two. The keys could be stored, for example, on the web or on a DNS-server. The customer willing to make a payment could get the right public key over the network. One problem is to find just the right key from a database containing the keys of people and organisations having the same or very similar names. This arrangement wouldn't also be economical in transaction time. Moreover the protection the database from attacks would be a problematic. Therefore, a good way of providing the public key is to get them from the merchant in the course of the transaction. However one must be sure that the key offered doesn't belong to a swindler. For that purpose there can be a trusted third party guaranteeing that the key offered belongs to the merchant. To prove this the merchant gives a certificate signed by the third party where the public key and the identity are connected.

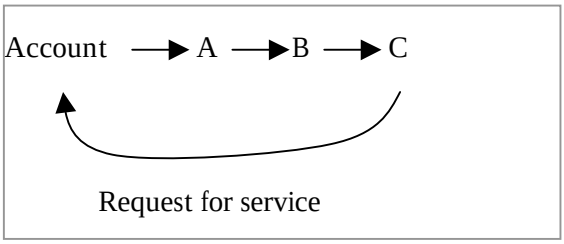
However, the use of a trusted third party poses a thread to an unalerted customer. The attacker could make up an own third party organisation and get all the guarantees needed. There is no fixed amount of third party organisations, which makes it impossible not to let the user to accept an affirmation certificate from an unknown party. Our solution is based on the assumption that in the future there will be a secure way of getting the public keys required.

Connecting the certificate chain

The bank must get the entire certificate chain from the bank to the party asking service to decide whether to provide it or not. In one certificate chain there can be several certificates which all can be in separate places. See the

figure 1 for an example. The customer C gets a certificate from a person B. The certificate grants rights to use the bank account of a person A. Now, the bank can't provide service on the basis of just the B-C certificate. For C to get service from the bank, the bank should get somehow the two missing certificates. The bank probably has the certificate granting account rights to person A, but the A-B certificate must be searched. Who is responsible of connecting the parts of the chain i.e. searching the missing certificate. Should it be the bank, the merchant or the customer?

Figure 1



First, we could obligate the bank to search the missing parts of the chain. However, this would open a possibility of performing a denial of service attack. An intruder could offer huge amount of fake chains to the bank server and ask for the server to look for the missing part that doesn't exist. Therefore this option is not an optimal one.

Second possibility is that the merchant would be the one connecting the lose ends of the chain. Thus the load of the banks would decrease. However, the merchant can also be busy. Therefore the customer should also be capable of providing missing certificates.

So, third, we can decide that the customer is responsible of providing all parts of the chain. There are two possibilities to do this. First alternative is that the customer must get the missing parts of the chain from a server. Other possibility is that the customer must keep all the parts of the chain in the memory of her terminal. Both solutions have their own benefits and shortcomings. If the customer has the complete chain ready in her terminal, the transaction process is lighter. That's because looking for a certificate from an online server costs time. On the other hand, multiple long certificate chains in a terminal increases the amount of memory required. If the customers would need to provide only the last certificate of a chain, the client side of the system could be thinner and more certificates providing access to different resources could be stored to the terminal. The memory demands are essential to the functionality of a mobile terminal.

That's the reason we propose that the customer should be the primary provider of certificate chains. There could be a possibility to ask for the bank to do the completion of the chain if the customer couldn't do it. The bank should still always have the right not to provide service if overloaded.

Online checks

The main function of online check is to ensure that any of the certificates in the chain isn't revoked. It can also be used to control the times a chain is used. One could have a bus ticket certificate containing four trips. Every time the chain would be checked on the online server, the count would increase by one. The fifth time the chain would be offered the check wouldn't go through any more. This can't be achieved with the certificates only. That's because the certificates can't count how many times they have been used.

An owner of a resource isn't really enforced to give service, when a valid certificate chain is offered. In exceptional circumstances a bank could refuse to perform a transaction without any external reason. The purpose of online checks is to assure the bank about the validity of the chain in order to get the service. To be sure that the chain is valid, the bank will postulate online checks at least in situations where the amount of money transferred exceeds some limit value. Online checks will slow the transaction process. Therefore it isn't a good practice to always demand online checks. On the other hand the merchant is taking a risk every time when not making the check.

Micropayments

An issue largely debated is the possibility to charge small amounts of money in the Internet. We are aware that our solution isn't optimal for very small purchases. That's a result of overhead cost (in processing time) of every transaction. One must first get the public key of the service provider in a secure manner, to make a certificate and send it to provider. No online checks are needed due to small amount of money transferred however. The reasonable

lower limit of transaction can be defined after evaluating a complete implementation of the system.

Solution

There are multiple phases in the process of using digital bank certificates. First, the bank customer must apply for the primary certificate that the bank can link to the account. Then that certificate can be delegated and finally some certificate is presented to a merchant offering services. Let us now go thru these steps and see what happens in each.

Acquiring the primary certificate

First, we assume that the user has an account in a bank that supports certificates. Secondly, a key pair or "identity" which is to be authorized is required. The client must have a secure way of generating the key pairs, and we feel that the most secure way is by generating the keypairs by self, in the customer's personal computer or PDA. The reason for this is, that should any party gain access to the customer's private key, it could pretend to be the customer.

To acquire the certificate, the first thing that needs to be done, is contacting the bank in a secure way. We propose a network connection with ISAKMP provided security, but naturally the client can also walk into the bank with his PDA, or even a floppy disk.

Now, the bank needs to know that the customer really has the permission to use the account in question. Since we are not trying to remain anonymous to the bank, this is done by authenticating the customer. Over a network, this can be done using a trusted PKI, in Finland that would be the FINEID (Finnish Electronic ID) card that provides identification certified by the Population Register Centre. However, the method of authorisation can be decided by the bank. Also, we assume that the bank can authenticate itself to the client using a similar PKI or by the customer physically walking into a bank office.

Now the client tells the bank the account that is to be linked to the certificate. The bank takes the public key provided and links it to the account. The key space should be large enough to avoid collisions, but the bank may choose to add extra security and check that the key is not previously used. Providing the authorizations in the certificate are acceptable, the bank signs the certificate with its private key, the public counterpart of which is publicly and widely distributed since the bank is not trying to remain anonymous. The certificate contains the public key of the bank (issuer), the public key provided by the client (subject), a bit signifying that the certificate can be further delegated (delegate), the actions authorized (authority) and the conditions of validity like a period or an online check (validity). It is significant to note that the account number the certificate refers to is not to be found in the certificate, since that would ruin the anonymity. A signature is also an essential part of the certificate, it certifies that it was indeed the bank that issued this certificate and not some impostor. The customer can now use this certificate to pay for purchases.

Delegation

The primary certificate is only usable by the primary customer. He might want to give somebody, say an offspring or a sibling, a right to use his account, perhaps for a limited sum or duration. To delegate a certificate, the public key of the recipient is needed. Here we again run into the problem of finding the correct public key. The delegator needs to be certain that the public key to which he is delegating really belongs to the correct recipient.

We can again use ISAKMP to provide us with a secure connection and FINEID or similar for the identification of the recipient. The public key could also be certified by using PGP or similar system, providing both parties are using it.

Once the public key has been transmitted and assured to be correct, the delegator will write a certificate with the desired authorities. These can be the same set or a subset of the authorities possessed.

In practice, these authorities can also be a superset, but unless the end client can present a chain that provides such rights throughout the chain, the wider authority is not granted. Basically this allows an issuer to delegate rights that he himself does not yet possess, but knows he is going to get. However, a certificate with the extended authority must be available for the forming of the authority chain at time of use.

Finally, the conditions of validity must be set. We use short time certificates with different public keys to protect the customer's privacy and therefore there is usually no reason to issue a period of validity that exceeds that of the parent certificate. Other conditions of validity can also be set. We believe that these will usually be some sort of online checks. There is no reason to declare any specific standard for the conditions of these checks, but the query interface should be standard.

This allows for the implementation of a variety of checks for a variety of applications which brings out the main benefits of the system. We can limit the amount of money that can be used, the number of times it can be used, or even more complex conditions.

We predict that there will be a set of standard checks available on specific online check servers for the end user's personal needs, in addition to checks provided by the service in question.

Purchasing

Purchasing is in fact just a sort of delegation that is combined with the act of transfer. The client delegates the merchant the right to make a single money transfer of the billed amount.

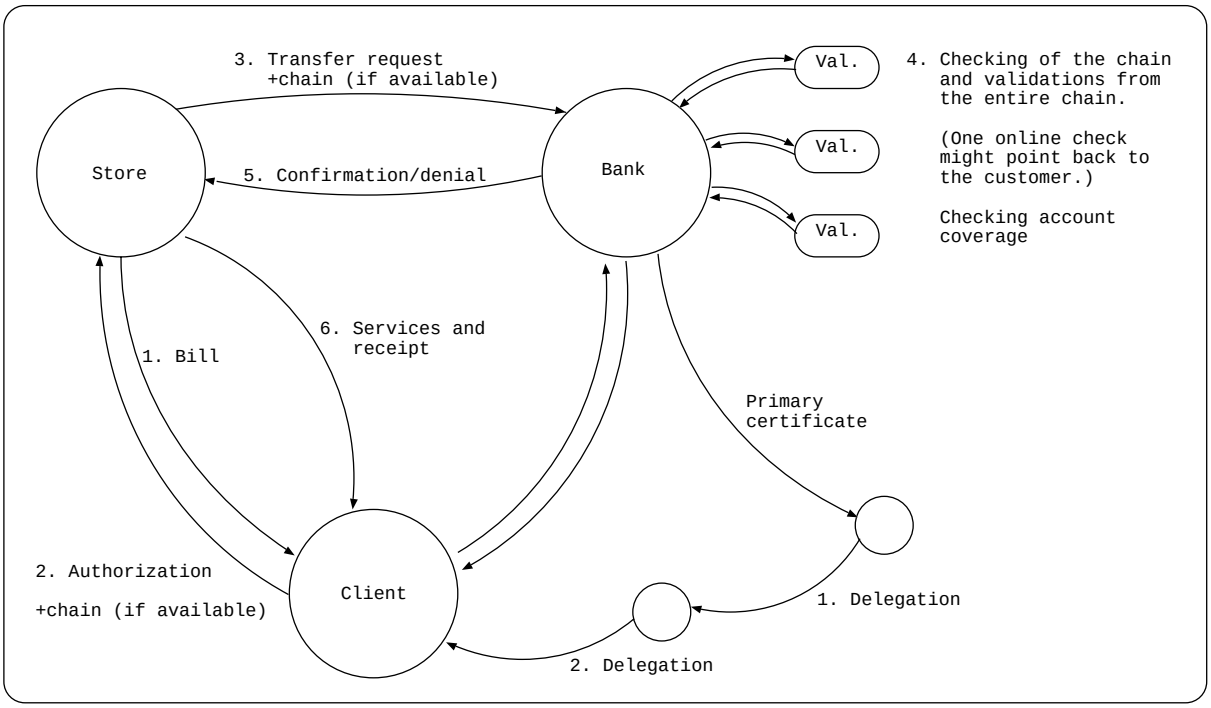
When we are emulating a bank card or cash, the money transfer should take place immediately, so that both the customer and the merchant get verification that the transaction really completed. When emulating a credit card, the money transfer from the customer's account happens sometime in the future. The bank, or some third party like a credit company, would make a reservation or note for the money, charging it at the correct time. Since the system allows these charges to be at seemingly random times, the customer needs to be aware of the charges to come. That problem, however is outside the scope of this paper, we only note that the bank could provide a way to check coming charges that are known to it.

Now let's take a look at what happens when a customer attempts to make a payment at a store, whether physical or network store. Again, we need a secure line of communication with the merchant. Therefore we can rest assured that no outsider can listen in on our business, and that the party we are dealing with is the correct one. For this we again propose ISAKMP and some PKI, but other methods can also be used.

In a cyberstore we might negotiate the channel before desiding what service we want, but we bypass that since we are only interested in the payment system.

Therefore, when it is time to pay, we assume the merchant has been authenticated and a secure channel is in place. The merchant now sends the client a bill (step 1 in figure). The software presents the customer with the price and writes a certificate for the sum. The customer now has to authorize the certificate for it to be valid. The certificate is signed and passed on to the merchant (step 2). If the terminal the customer is using has enough memory to store the entire chain of certificates, it should do so and pass that to the merchant as well. The merchant verifies the signature on the certificate and, providing it matches, writes a certificate for money transfer to be sent to the bank, possibly with a chain (step 3).

The bank is generally the party that assembles the chain, so it checks whether the chain provided is incomplete. If it is, the missing parts must be collected. In order to secure the bank against Denial of Service attacks, the bank can refuse the collection of the parts. In such a case it tells the merchant to present an entire chain. In case the merchant is also busy, such as a popular network store, it can again refuse the task and tell the customer to handle it. We note that in a conventional physical store the merchant is likely to have enough resourses for the task since a cashier handles customers one at a time. Furthermore, the network connection of the customer in a physical store is likely to be slow and go thru the network of the store. Therefore, the merchant would likely be a more efficient searcher for the chain. In a network store the situation is different since the server handles multiple requests simultaneously. However, now the client is likely to have a relatively fast network connection and more processing power, allowing the client software to collect the chain.



Eventually someone has collected the necessary certificates and the chain is in the possession of the bank. The bank now takes a look at the certificates in the chain and checks that the authority required is valid. Next the bank checks for the online checks in the chain. All the limit type checks are performed using two phase commit[15]. In the first phase the bank asks the servers whether there are usages left and reserves the resources. Provided all the checks have resources left and can be reserved, the bank commits all the reservations and makes the transfer (step 4).

Further usefulness of the online checks can be added by making the online check of the payment certificate back to the customers PDA and the client software. This forms the payment chain into a complete circle, proving to the customer that the commits have been performed by the bank. This feature is quite useful when simulating cash, since the money transfer takes place immediately. In more credit like situations we feel that the checks should be made to a server, since there is no guarantee that the customer's terminal is online at the time the money transfer takes place.

After the bank has transfered the money, it returns a digital receipt of the transaction to the merchant (step 5) who then can provide service to the customer and possibly also write a receipt, whether digital of physical (step 6).

Future work

Combining the certificate-based paying concept with smart card technology is interesting because the key pairs needed to form certificates need storage. A smart card would make the system independent of the media through which the customer likes to make purchases. However, the limited memory available on smart cards poses a limit on the number of certificates that can be stored. Also mobile applications working inside a mobile phone or Palm™ could be constructed. Such devices also have more memory available, therefore a combination of smart card and a PDA might prove to be a realistic option. Especially since using the smart card in a non-trusted terminal is highly risky as the customer cannot be certain what the terminal actually does with the PIN code.

Also further development of the system would be required for it to be considered really useful. Usefulness of the system could be evaluated by testing and optimizing to find out how small a purchase would still be worth performing.

One interesting lead to follow would be the development of electronic money and it's impacts on network payment.

Conclusions

In this paper we discussed the problems of current electronic payment systems such as the customers insufficient anonymity and control of transaction process. We also did set criteria for a desirable system from four point of views: customer, system, security and other. We proposed a new system based on chaining SPKI authorisation certificates. We argued that by using this system we can tackle many problems concerning anonymity and the customers defenceless against the merchants rights to use customers account.

References

[1] White, L. H., The Technology revolution and Monetary Evolution. In: Dorn, J. A. (Ed.), *The Future of Money in the Information age*. Cato Institute, 1997. http://www.cato.org/pubs/books/money/tableof.htm

[2] Maksukorttitietoa kaupalle: Varmennus, (Finnish instructions about checking the identity of a credit card user), Luottokunta, 1999. http://www.luottokunta.fi

[3] Addressvitt Services, 1996. http://www.addressvitt.it/gballser.htm

[4] *Micropayments overview*, W3C. http://www12.w3.org/ECommerce/Micropayments/Overview.html

[5] *Micropayments' future uncertain*, Reuters, CNET, April 17, 1998. http://www.news.com/News/Item/0,4,21198,00.html?st.ne.bp..bphed

[6] Crocker, S., *The Siren Song of Micropayments*, April 22, 1999, iMP Magazine. http://www.cisp.org/imp/april_99/04_99crocker

[7] *GVU's 10th www User Survey*, October 1998. http://www.gvu.gatech.edu/gvu/user_surveys/survey-1998-10

[8] BW/ Harris Poll, *Online Insecurity*. BusinessWeek, The McGraw-Hill Companies Inc, March 1998, complete survey at BW-online: http://www.businessweek.com/@@WZJy4cASJ*2SwAA/1998/11/b3569107.htm

[9] Medvinsky, G. & Neuman, B. C., *NetCash: A design for practical electronic currency on the Internet*. Proceedings of 1st the ACM Conference on Computer and Communication Security, November 1993. http://nii.isi.edu/info/netcheque/documentation.html

[10] Neuman, B. C. & Medvinsky, G., *Requirements for Network Payment: The NetCheque Perspective*. Proceedings of IEEE COMPCON'95, March 1995. http://nii.isi.edu/info/netcheque/documentation.html

[11] Peirce, M., *Payment Mechanisms designed for the Internet*. http://ganges.cs.tcd.ie/mepeirce/Project/oninternet.html

[12] Ellison, C. M., Franz, B., Lampson, B., Rivest, R. L., Thomas, B. M., Ylönen, T., S*imple public key certificate*. Internet draft (expired), IETF SPKI Working Group, March 1998.

[13] Lehti, I., Nikander, P., *Certifying Trust*. Public Key Cryptography—First International Workshop on the Practice and Theory in Public Key Cryptography PKC'98, Pasifico Yokohama, Japan, February 1998.

[14] Schneier, B., *Applied Cryptography 2nd edition*, Joahn Wiley & Sons, 1996.

[15] Kortesniemi, Y., Hasu, T., Partanen, J., A Revocation, Validation and Authentication Protocol for SPKI Based Delegation Systems, To appear in: Proceedings of the 2000 Network and Distributed Systems Security Symposium, February 2000, San Diego, California, Internet Society, February 2000.

[16] Davida, G., Frankel, Y., Tsiounis, Y., Yung, M., *Anonymity Control in E-Cash Systems*. Financial Cryptography Conference 1997, (FC97), Feb, 1997.

[17] Saarela, J., Mechanisms of Electronic Money. http://www.tcm.hut.fi/Opinnot/Tik-110.501/1995/ecash.html