Payment and privacy: a key for the development of nfc mobile

Jean-Claude Paillès, Chrystel Gaber, Vincent Alimi, Marc Pasquet

Payment and Privacy: A Key for the Development of NFC Mobile

Jean Claude Paillès, Chrystel Gaber, Vincent Alimi, Marc Pasquet
GREYC laboratory
ENSICAEN-University of CAEN-CNRS
jc.pailles@voila.fr

ABSTRACT

This paper first introduces a possible evolution of secure personal identification devices, based on RFID technology in the mobiles phones (NFC). Given the characteristics of the mobile phone market, this trend could grow quickly and importantly. This paper considers the possible impact of this evolution in term of privacy, focusing on a typical and important case: payment transactions. This paper sticks to the general approach and role of "card payment system". Yet, it demonstrates that it is possible to improve some of the privacy characteristics of this kind of application. It also outlines the way payment protocol should be designed in order to reach this goal.

KEYWORDS: security, payment, EMV, anonymity, mobile commerce, NFC, privacy

1. INTRODUCTION: NFC MOBILE AND PRIVACY

Today, mobile phones are used to make phone calls, send text messages, photos or short movies, visit web sites, send emails, watch TV, buy items (mobile payments), take the train (mobile ticketing), access to a company's intranet, etc. The mobile is becoming a central tool in its owner's life.

This trend is amplified by the introduction of NFC technology [1] and by the Internet browsing capabilities, which are now available on most mobiles. These evolutions enable mobiles to be used for many transactions in the daily life, either close (through NFC technology) or distant (through mobile network technology such as GSM/UMTS). Mobile will become the "Swiss army knife for day to day transactions" as shown on the drawing below. However, if all the owner's (user) actions can be traced and linked together, it would lead to a major threat for his privacy.

Hence there is a strong need to provide the user with the guarantee that his or her privacy is protected by preventing third parties to link together his different actions. Of course this issue on privacy appears also for many transactions systems, whether they are smart card based or not. However, we put the stress on mobile payment because such systems are not yet deployed, and it is therefore not too late! Moreover, it may be considered that deployment will happen very soon. It could be massive, and could cover various types of applications, like payment (most probably a killer application), as well as loyalty, ticketing, access control, etc, using the same personal item, ie the mobile phone. All that could lead to a dramatic privacy issue.

Figure 1. NFC Mobile Phone a Swiss Knife?

Often, contactless technology is considered to be more sensitive to privacy issues than usual contact technologies used by classical smart cards. In the case of NFC-mobiles, a transaction will be triggered by the outside world only if the mobile owner has previously selected the corresponding application in a menu. This user's positive decision is similar to a classical smart card which has to be inserted in a slot. However, unlike classical contact technology, an NFC link may be easily spied on without tampering with the reader device. Here we make the hypothesis that the entities attempting to invade the user's privacy may be the same than the parties involved in the transactions (Banks, MNO,

Merchants, clearing systems, etc) as well as outsider parties. The proposed study covers both cases.
The linking of the transactions and the user may be possible if during transactions, the mobile sends to the merchant terminal or server some data related to user: identity, certificate, card number, customer number…etc[1]. Thus, privacy enhanced transactions must be designed in a way that they release as little personal information about the user as possible, without, of course, diminishing the application's security. This excludes the use of classical SKI or PKI cryptography for implementing the security required by applications. Within this context, the cryptography methods to be used have to enable some level of anonymity. Annex gives a reminder of such techniques. The FP6 EU project PRIME has explored these issues [9], and IDEMIX [10] is an industrial product related to this work.

However, there is also a need to allow some kind of user's traceability to counter misuses of a service and to allow the possibility to revoke users in case of attacks resulting for example in the leakage of cryptographic material (keys). Hence, although user privacy is important, we often see that complete user anonymity is not desirable and that some level of traceability is necessary for revocation purpose. As a proof of this statement, the reader could look at the TCG (Trusted Computing Group) standards [3] where a specific function for anonymity has been specified, the DAA (direct anonymous attestation) based on the ZKPK cryptography (zero knowledge proof of knowledge; see references [6] and [8])[2].

2. PRIVACY TECHNIQUES FOR PAYMENT

Privacy is important for any kind of transaction: ticketing, access control, etc. Worse, all these kinds of transaction could be supported by the same mobile phone.

We focus our contribution on payment, because it is the basic, the most "always used" and the "killer" application, and because it is also a very sensitive function with high-level security needs.

Preoccupations with this issue started a long time ago, when the model of anonymous electronic coins was first developed by David Chaum [2]. This theoretical model is however very difficult to roll-out as the main issue is the risk of double spending. Different schemes have been proposed in various papers. They try to address the problem of coin divisibility (how to pay ½$ with a coin of 1$) as well as the problem of transferability (how Alice can give to Bob a 1$ coin keeping security, i.e. preventing any money creation).

However, it seems that this coin approach leads to complex protocols, and no real implementation seems to exist now. A lot of successful research has been devoted to divisibility but the problem of double spending remains difficult, and requires an on-line verification during transaction by a central authority. Practical roll-out of these approaches is difficult especially when small expenses are targeted by these payment systems.

The concept of electronic purse appeared 15 years ago, and is based on the idea that it is possible to secure a money provision in a handheld device, like a smart card. Transaction is simply a secure exchange of money between the client smart card and the merchant secure device. Security here needs mutual authentication between the two devices. There is no need to collect the client purse identity.

However, most electronic purse systems try to prevent card cloning by monitoring the expenses related to one purse id. Even with smart cards, it is difficult to prevent any physical attack potentially leading to a multitude of clones. Since an attacker may obtain important financial benefits from cloning, he may use specialized and expensive expertise or technologies in order to perform such attacks and to produce a multitude of clones of the same card. Therefore, most "serious" electronic purse systems also use individual transaction collection as means to detect and blacklist abnormal consumptions.

Privacy for electronic purse without individual transaction collection is easy to obtain. For example, if the user always pays the reload of electronic money in its purse with cash, there is no possibility to further trace and link the transaction done by this user. Instead

---

[1] The low layer protocols for NFC were designed in such a way that they need some kind of constant identifier, thus enabling to trace all the transactions coming from the same mobile. Fortunately, the ETSI standard TS 102 622 has defined a mode where this id (necessary for collision issues in contactless transmission) may be chosen randomly at each transaction.

[2] The reader will see that a specific element called "lambda" has been added to the DAA protocol for solving exactly this issue.

of cash, some form of electronic coin (like introduced above) may be used to obtain the same result..

Privacy for electronic purse with individual transactions collection has the same kind of solution as the one described below for the card payment model.

So far we have focused on face-to-face payment, without addressing distant payment for e-commerce. The classical "card payment" model, where the client gives his bank-id to the merchant or to a trusted 3^{d} party, is universally used, with a well-known lack of security.

Systems like the "3D Secure" [11] bring much more security with a better client authentication. Regarding privacy, those systems don't really care about preserving any client anonymity.

3. THE «CARD PAYMENT» MODEL

3.1. Principles

The principles of this kind of payment model, which is now fully rolled out and used in many countries in Europe, are simple. They are described below on Fig 1. It shows the mains steps of payment transactions, with the possibility for mobiles to follow two paths for the authorization step. The first path goes through the merchant terminal, whether the second goes directly by the mobile to the issuer server, by sending short text messages for example. Payment and authorization messages comply with the so-called EMV payment standard [4]. Current implementation use smart cards as the device supporting these payment transactions.

Contactless is becoming a trend for smart cards as well as mobile phones, which use the "NFC" standard. Thus, mobile phones could become in the near future a support for payment transactions. Several projects are working now on this issue such as the French project "PEGASUS" [1]. However, this does not change the general descriptions provided in this paper which covers the two kinds of implementation. The word **"card"** used below stands either for the **smart card** in a smart card implementation or the card payment application inside the NFC mobile phone, which is located in a **"security element"** which can for example be the **SIM**.

In the drawing below, the Client has a smart card or a mobile phone with NFC contactless capabilities.

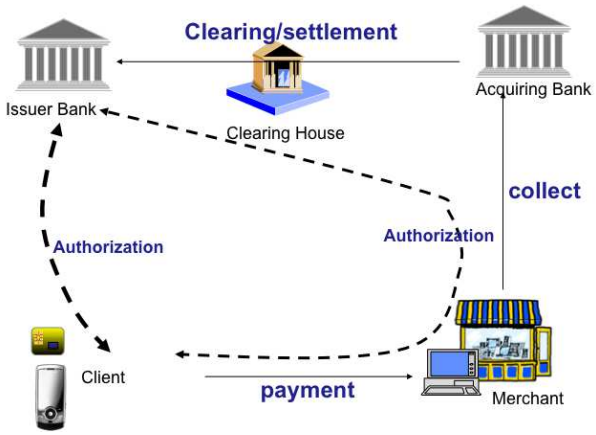


Figure 2. The Card Payment Model

The EMV standard supports possibility of off-lines transactions for better performance and reduced costs for Merchants. The security for protecting bank or merchant of excessive spending is enforced through different means. An example is an on-line end-to-end authorization process between the payment card and the Issuer bank server, which gives to the payment card an amount limit for subsequent off-lines transactions. This process is called risk management.

The card performs the following different steps:

- Client authentication by Merchant POS, in order to resist to card or device theft.
- Risk management for protecting Issuer Bank from excessive spending by the client.
- Eventually, the authorisation process, detailed below.
- Transaction execution leading later to transaction finalization.

Each of these elementary steps may be detailed as follows:

- Client authentication :

It may be a simple PIN control or biometric check.

- Risk management can for example be a consumption ceiling loaded by the Bank in the Client device associated to some date limit. At each transaction, the card will perform risk management, i.e.:
  - Verify that current date fits with date limit.
  - Verify that ceiling is sufficient with respect to the transaction amount, and if yes, update the ceiling by decreasing the amount
  - If not, an authorisation exchange takes place between card-Merchant POS-Issuer, and normally, a new ceiling will be reloaded in the card, else the Issuer refuse the payment, which terminates the transaction

- This authorisation exchange is of course very sensitive, and end to end security protocols are performed between card and Issuer[3].

- Transaction execution: the transaction is executed. A set of relevant data and a proof are then given to the merchant terminal (POS) by the card. The proof is a signature of data like Card Id, Merchant Id, Timestamp, amount, currency, etc. The client and merchant ID contain their associated bank ID.
  - This proof implies that the card is genuine, and this step covers card authentication.
  - This signature has to be verified by the Merchant POS. Using a SKI approach, it would be necessary to have a secret master key in the POS, in order to re-compute the card signature key, and verify the card signature. This solution would be very security sensitive and master key compromise would be catastrophic for banking institutions. The chosen solution is PKI based, and is much better in this case, since only public keys in the POS are necessary[4].
  - Merchant POS stores the transaction data in a file
- Finalization:
  - The transaction data file is collected periodically and sent to the acquiring bank (i.e. the bank of the merchant)
  - The acquiring bank credits the merchant account accordingly
  - Clearing/settlement process happens periodically; clearing center receives the transaction data from all the acquiring banks, sorts it by issuing bank, and sends it to these banks
  - Issuing Bank verifies the transaction certificate (TC), and debits accordingly the client accounts.
  - In some cases, (i.e. client account is empty) a charge back may be sent by the Issuer's bank to the acquiring Bank, since it may happen that the risk management has not been sufficient (case of a client paying with different means of payments: card and cheque for example). Yet, The regulations orientation is towards more responsibility to the issuing bank, thus charge back may disappear soon.

3.2. Privacy Aspects

Of course, payment systems have to securely enable the debit of the client account and the credit of merchant account of the same amount. This means that the payment system has to prevent or at least has to enable detection and revocation of all classical attacks: fake cards, replay of transaction, even considering collusion between for example one Client and one Merchant employee, etc. Actually this goal is reached by existing smart cards payment systems, but they pay a high price in term of lack of privacy. Merchant or banks obtain all information regarding Client transactions: **such Client**, known by its Bank identity (i.e. Card number) has performed **such payment**, **such date**, with **such merchant**, and for **such amount**. Moreover, **they get all necessary data to prove this statement**, like certificates and electronic signatures.

Enhancing privacy means that no entity (bank or merchant) is able **to know and to prove**[5] a statement like the previous one. It is possible to organize a card payment system respecting this principle by separating the two following operations and related data:

- Debit of the Client
- Credit of the merchant,

while respecting the two constraints below:

- A transaction must lead to one debit of a client and one credit of a merchant of equal amount
- No linking is possible between debit and credit; however on some period of time, clearly it will be possible to control that Σ debit=Σ credit

These constraints will be satisfied with the security principles appearing on figure 3. It shows the separation between "debit and credit tickets" delivered by the mobile phone to the POS. Their authentication has to use special signature functions in order to prevent any linking of the data they contain. Details are given in section 4.

[3] Two possible paths may be used for authorisation message exchanges which can spend a few seconds: through POS in case of contact smart cards, or OTA with NFC mobiles, where NFC session must be short (<0,5 s) at least for the response of the authorisation server.

[4] In fact the signed data contain a so called « transaction certificate » which is a MAC of other transaction data, and which is sent to the Issuer after transaction collection, and then verified by it.

[5] What is meant by proof is for example if a message with its signature and signer certificate is found, then it is a proof that this message has been signed by such person or body.

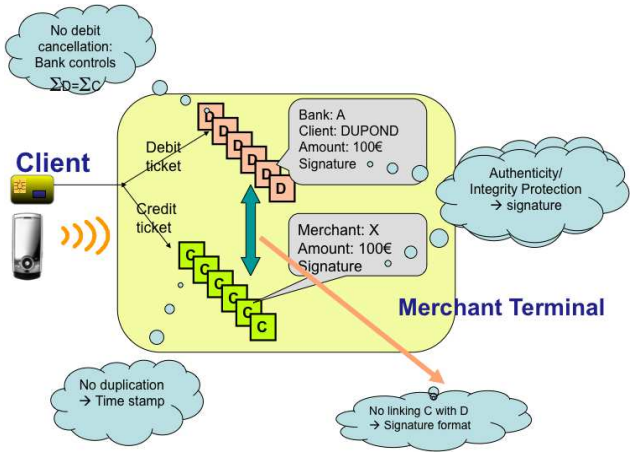


Figure 3. Satisfaction of the Security Principles

4. A MODEL FOR A PRIVACY ENHANCED CARD-PAYMENT SYSTEM

4.1. Elementary Data

Here are the basic data our protocol has to manage.

- *m*: merchant Id
- *c*: Client Id
- *b*: Issuing Bank Id; b+c can be considered as the so called PAN appearing on the credit cards
- *t*: timestamp
- *a*: amount and currency code of the transaction
- *c'*: hidden *c* (by encryption)
- *s1, s2*: 2 signatures;

Usage details of these elementary data by different processes will appear below. The role of *c'* is to prevent the PAN to be known by the Merchant.

4.2. Transaction Process Principles

The figure 4 shows the different steps, which are performed from payment transaction on the Merchant POS up to the clearing and settlement between banks. Weak security link is at POS and message transmission level, since the links between banks and processes in bank servers may be considered as trusted.

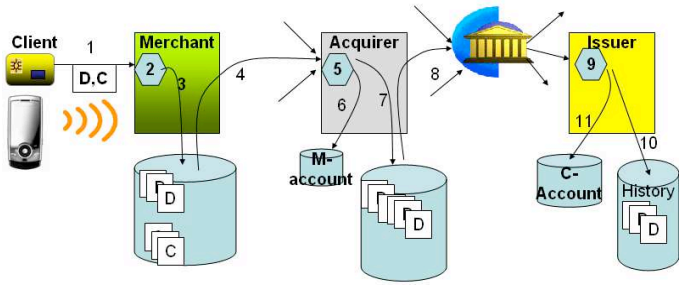


Figure 4. The Different Steps in Payment Transaction

1. The transaction performed on the client's side may be organized like in EMV for client authentication and risk management. It differs for the content and format of data delivered at the end of the transaction to the POS terminal, since it has to deliver the two pieces of information, D & C:

*D=b,c', a, t,s1 (*s1 signs previous data of D)

C=m, a, t, s2 (s2 signs previous data of C)

The merchant must not have any means to prove that such D relates to such C, i.e. both come from the same payment device, except for the "*a*" data, but which is not considered here as a real mean to link and prove that two pieces C and D correspond to the same client.

s2 is important for merchant, to verify authenticity of the payment device, but this signature must not enable any linking between the two: so mechanisms like group signature are required here. See annex 1. In short *s2* is a proof that the C piece has been computed by a genuine card, without giving any indication on the particular card (ie its identification) which has been used.

s1 is only used by Issuing bank to verify integrity and proof of origin of D part; *s1* may be a simple secret key signature, (as a message authentication code MAC, like in EMV) since Issuing bank has issued the card and the secret key it contains, and thus knows the key.

t is a timestamp for detecting replay attacks: it is chosen by the POS in such a way that:

- on this POS, two transactions can't use the same timestamp value *t*
- on a set of POS (like those collected by an acquirer) which clocks are not precise enough nor synchronized, it is possible to have different transactions using same value for *t*; see remarks below

Note that having a record D and a record C with the same *t* doesn't allow merchant to reveal and prove that such client has performed such transaction at such date, since D contains *c'* and not *c*. In fact, the Merchant can exhibit sound records D and C having

the same *t*, and same amount *a* but it is impossible for the Merchant nor the Acquirer to prove that these two records correspond to the same Client.

2. The merchant verifies D and C, ie verifies the signatures *s1* and *s2*, and of course that *a* (amount) and *t* time stamp are the same in D and C
3. POS stores D and C in mass storage, in order to perform later the collect operation using the file of D&C records.
4. The collection of several D & C records is sent to acquiring bank.
5. The acquirer has to perform some controls: we consider that the acquirer doesn't trust the merchants. Thus, it has to protect itself against frauds like inventing fake records D or C, or removing some C record in case of a collusion between a client and a merchant which would allow the client not to pay.

So, the acquirer has to:

- Verify that C & D are well formed and number of D = number of C
- *Verify s2* of C records (which doesn't give any information on who is the client, as said before).
- Verify that neither D nor C is duplicated: i.e. No two D or C with the same t for the same merchant.
- Verify that $\sum_{DC} a = \sum_{DC} a$ i.e. verify the balance of debit and credit collected

The collect of the merchant is rejected if something goes wrong.

6. The acquirer credit merchant account of $\sum_{D} a$
7. The acquirer stores D records in mass storage.
8. Each acquiring bank sends to a clearing center the D records which are in mass storage, and removes it from its mass storage

The clearing center sorts all the D records by *b* (issuing bank Id) and sends it to the relevant issuing bank

9. Issuing bank performs:
   - *c'* decipherment and obtains *c*
   - verifies *s1*, which may be a simple SK signature
   - verifies that no 2 identical D records have been received for this client, using the history file
10. stores D record in the history file
11. debit *c* client account of amount a

4.3. Securit10y and Anonymity Properties

Security: the following classical attacks are covered:

- A client can't forge a fake transaction with the D and C records if the signature keys used for computing *s1* and *s2* are not compromised. This issue relates to the security robustness of the secure element in the mobile performing these computations.
- The merchant can't create from scratch a fake transaction with D & C records, since controls of the Acquirer would detect it.
- Collusion between Merchant and Client is also impossible, for the same reasons
- Anonymity: is a Bank, Issuer or Acquirer, able to prove that such client has made a payment of such amount, such date, for such Merchant?
- According to § 4.2-1 *s2* signature doesn't give any information on the signer. The amount is not considered as a link between the C and D records as said before. Timestamp t is discussed below.

Remarks

a/ Timestamp *t*

Its precision has to be chosen carefully, in order to have good properties for privacy while keeping good security against some kind of replay attacks. Let's take an example:

- If the precision of t is one day: a Client could use a D from a previous transaction performed the same day on this POS or another POS in order to pay twice but being debited only once
- if the precision is one micro-second, collusion between banks (issuer and acquirer) would enable to link together D and C records corresponding to the same transaction, because probability for different transactions to get from the POS the same t would be very low.
- Thus, a trade-off between these 2 extreme cases has to be chosen to keep both security and untraceability. This trade-off has the following characteristics:
  - It doesn't give a mean to link the 2 pieces D & C, and to prove this linking
  - It prevents replay, i.e. it must not be possible for a client to provide a POS with C (meaning that the merchant will be paid of a) and a D of same amount *a* but corresponding to another client, in order for the client not to be debited.
  - If D is the minimal time between two transaction on the same POS, *t* must be precise at <D/2, for example D/3

b/ Cryptogram c'

c' is a cryptogram of *c* which has to be recovered (decrypted) by the issuer bank. Thus, there is no big issue here, and the Issuer may use any mean to establish

correspondence between c and c'. This correspondence can be made variable with time, according to the same algorithm on the two ends: security element in the mobile and Issuing Bank

c/ Cloning Detection.
As said before in § 2, cloning has to be considered as a credible threat if no detection/revocation capabilities are possible. Here detection capabilities remain because in step 10 & 11, the bank will see an abnormal consumption of a particular client (whose card has been stolen and cloned for example). Revocation is easy by a classic blacklisting capability of merchants terminals, based on the data *c'*. Other possibilities appear with mobiles, since it is easy for example to update all keys or certificates of all mobiles except for the mobile containing the SE which has been cloned.

d/ Charge Back
This function would require relating a C and a D record, while the main goal of this paper is to prevent this association! This issue, if considered as important could be solved through the use of some special kind of anonymity scheme, called "conditional": ie it is possible to a specific authority to break anonymity.

5. CONCLUSION

The development of NFC technologies gives mobile phones the capability to be used for many transactions in the day to day life, proximity (through NFC technology) or distant (through mobile network technology). The mobile will become the "Swiss army knife for day to day transactions". However, if all the actions of the owner (user) can be traced and linked together, it would lead to a major threat to the user's privacy, and "BIG BROTHER" is not far away!

Indeed, privacy is not the main issue for the development of NFC mobile phones technologies. The issues regarding the security of the mobile and its SE, or the constitution of an ecosystem between different operators and service providers, taking into account all the stakes involved and accepted by all partners are certainly more important.

However, we consider that taking into account privacy aspect in an early development phase of this new technology would be wise, since with mobile phones, we have already seen and we could see in the future that the consumer appetite for new mobiles together with the proactive distribution policy of MNO could result in a fast roll out of this new technology and result in a massive coverage of the population.

We hope that this paper demonstrates that technologies exist and will help to develop consciousness and realization of necessary actions around this issue of privacy.

ABBREVIATIONS

NFC: near field communication
MNO: mobile network operator
POS: point of sale, here the merchant terminal at the point of sale
SIM: subscriber identity module
SE: secure element; ex: the SIM
Id: used for identification
EMV: Europay Mastercard Visa
SKI / PKI: secret key infrastructure / public key infrastructure

REFERENCES

[1] NFC Forum: http://www.nfc-forum.org/home. Contains a lot of documentation on this new technology: specs, white papers

[2] David Chaum: see Advances in Cryptology - CRYPTO 1984 (1985) Blakley, George, Chaum, David

[3]: TCG: https://www.trustedcomputinggroup.org/home; "Trusted computing platforms, trusting computing group"

[4] EMV: see the site EMVco: http://www.emvco.com/ containing specs and white papers for Chip Based payment systems

[5] PEGASUS: see the site of the AEPM new name of PEGASUS: http://www.payezmobile.com/)

[6] Zero Knowledge protocols: ISO/IEC 9798-5 Standard. Part 5: Mechanisms using zero-knowledge techniques - Second edition, December 2004.

[7] Anonymous Attestation Using Blind Signatures: INSPIRED –EC FP6 project deliverable, 22/04/2005

[8]: DAA: Direct Anonymous Attestation: Ernie Brickell, Jan Camenish, Liqun Chen: see http://eprint.iacr.org/2004/205.pdf

[9] PRIME EU-FP6 project: https://www.prime-project.eu/

[10] IDEMIX: see the site of IBM-Zürich: http://www.zurich.ibm.com/security/idemix

[11] Verified by VISA: http://www.visaeurope.com/merchant/handlingvisapayments/cardnotpresent/security.jsp

Annex: an Example of Cryptographic Technique for Privacy

In §4, we have seen that the signature S2 needs some "good" properties in order to cover the needs we have defined for card payment privacy. Several possibilities exist, for example "blind" signatures [2] or ZKPK protocols. Several European projects have also addressed this topic: INSPIRED [7] or PRIME [9]. A full description of ZKPK can be found in [8], and a possible usage and implementation is defined by TCG in [3] for the particular SE which is the Trusted Platform Module. Here is a short summary.

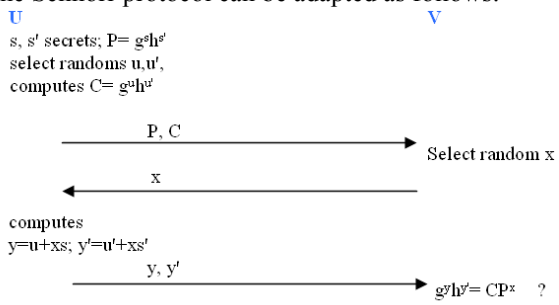
The discrete logarithm problem can be generalized:

- To the case of having several well chosen generators: for example having g and h , and c, and finding x and y such that $c=g^x h^y$ in Zn**.**
- Where n is not a prime, but an RSA number, i.e. a product of 2 primes. In this case computing e-th roots in Zn is possible.

Starting from the SCHNORR protocol [4], ZKPK can be introduced as follow:

Proof that entity U knows s, s' such that $P= g^s h^{s'}$

The Schnorr protocol can be adapted as follows:



This kind of protocol is named a « proof of knowledge » of s and s' such that $P=g^s h^{s'}$, and notation used is :

PK {s, s'; $P=g^s h^{s'}$ }. In this protocol, u et u' have only a circumstantial role, for blinding values s and s', which are the secrets of the commitment P, since this protocol can be seen as a commitment of U to prove to V that he knows these values s and s' such that $P= g^s h^{s'}$. Variables u and u' are often called "blinding variables". More over, this is P which is important, not C, which explains the notation above, where C doesn't appear.

This mechanism can be applied to any number of commitments: for example:

PK { s, z, r, r', r'' ; $A=b^s g^z h^r$, $B=g^s h^{r'}$, $C=g^z h^{r''}$}

Proof that U knows a, b, d, and that d=ab

U has sent commitments $B=g^b h^{b'}$, $A=g^a h^{a'}$ and $D=g^d h^{d'}$. To prove that d=ab, D can be expressed by $D=B^a h^e$, with b'a+e=d' . So U can give:

PK {a,a',b,b',d,d',e ; $A=g^a h^{a'}$, $B=g^b h^{b'}$, $D=g^d h^{d'}$, $D=B^a h^e$ }

If d=ab, then U can perform this protocol. The difficulty of the discrete log problem can be used to prove that d=ab.

Proof that U knows $v=c^{1/e}$, c public value, without revealing e, a secret of U

Principle of the protocol: U « blinds » e by a random w, et will prove that he knows vg^w and w, such that $(vg^w)^e=cg^{ew}$; he must prove that the g exponent, z , is the product ew. For doing that he uses commitments Cw and C_z and prove that $C_z=C_w^e h^{r'''}$.

So, U performs the following protocol:

U select random w, r, r', r '' and r''' and computes : z=ew , $C_v=vg^w$, $C=cg^z h^r$, $C_w=g^w h^{r'}$, $C_z=g^z h^{r''}$; he sends Cv, and then performs :

PK {z,w,e,r,r',r'',r'''; $C/c=g^z h^r$, $C_z=g^z h^{r''}$, $C_w=g^w h^{r'}$, $C=C_v^e h^r$, $C_z=C_w^e h^{r'''}$)

Numbering the 5 last values from 1 to 5:

→Clearly, if U knows e and v, he is able to perform this protocol.

→If U' is able to perform this protocol, then he knows $c^{1/e}$:

Let $j=C_v/g^w$; then $j^e=c$ since 2+3+5 imply that z=ew, since $5 = g^{ew} h^{er''+r'''}$ and $2=g^z h^{r''}$

1 + 4 imply that $cg^z h^r= C_v^e h^r$, therefore that $C_v^e/g^{ew}=c$, therefore that $j^e=c$. CQFD

This protocol can't be used as is as a proof that U is certified by authority A which knows the factorization of n, and then can compute e-th roots over Zn. But it is the basis of TCG SIGN protocol

Simplified ZKPK Protocol

R, S, g, h are public elements of Zn, of great order

In this protocol, user U has first to perform a JOIN procedure for obtaining after strong identification a certificate from an Authority, i.e. An e-th root of an expression $E=1/R^f S^v$ where f is a secret chosen by U, and v is imposed by the authority (otherwise, it would be possible for U to choose f and u as multiples of e!).

In the SIGN procedure, U proves that he knows an e-th root of E, without revealing e nor u or v. So, User U proves that he is part of the group of users which have been certified by authority A, since only A is able to compute e-th roots.