Quantum Oblivious Mutual Identification

Claude Crépeau[1 2] and Louis Salvail[2]

[1] LIENS, (CNRS URA1327)
École Normale Supérieure,
45 rue d'Ulm, 75230 Paris CÉDEX 05, FRANCE.
e-mail: crepeau@dmi.ens.fr.
[2] Département d'Informatique et R.O.,
Université de Montréal,
C.P. 6128, succursale centre-ville,
Montréal (Québec), Canada H3C 3J7.
e-mail: salvail@iro.umontreal.ca.

Abstract. We consider a situation where two parties, *Alice* and *Bob*, share a common secret string and would like to mutually check their knowledge of that string. We describe a simple and efficient protocol based on the exchange of quantum information to check mutual knowledge of a common string in such a way that honest parties will always succeed in convincing each other, while a dishonest party interacting with an honest party will have vanishingly small probability of convincing him. Moreover, a dishonest party gains only a very small amount of information about the secret string from running the protocol: whoever enters the protocol with no knowledge of the secret string would have to enter this protocol an exponential number of times in order to gain non-negligible information about the string.
Our scheme offers an efficient identification technique with a security that depends on no computational assumption, only on the correctness of quantum mechanics. We believe such a system should be used in smart-cards to avoid frauds from typing PIN codes to dishonest teller machines.

1 Introduction

Weren't you worried the last time you typed your PIN (Personal Identification Number) to an unknown teller machine that it could be a fake and that its sole purpose could be to steal your PIN? According to a recent headline of the NY Times [26] maybe you should worry:

"ONE LESS THING TO BELIEVE IN: FRAUD AT FAKE CASH MACHINE"

The problem with current identification systems is that the customer must trust the equipment to which he types his PIN. It is completely trivial to modify a teller machine to memorize the PIN numbers that people type to it. PINs are meant to be checked, not given. (Consult [1] for an extensive study of frauds at teller machines.)

Of course, it is always possible to completely solve the problem of identification and authentication of messages by classical methods [9] that require exchanging passwords which length are proportional to the number of uses. Unfortunately, this is completely impractical: we want to rely on the existence of a *short* secret to check identity.

A similar approach has been suggested in a computational model through the construction of pseudo-random generators [7] or pseudo-random families of functions [17] which requires only short secrets seeds. These solutions make sense only in a context where we put computational restrictions on the participants. For powerful parties it is trivial to fake identities.

In the computational model, more sophisticated tools were developed for this purpose: Zero-Knowledge Proofs of Identity [15] introduced in order to provide means by which an honest party may convince another party of his identity in a way that cannot be replayed successfully to another party. This is true even if the verifying party tries his best to extract valuable information out of the proving party. Moreover, a dishonest party attempting to prove an invalid identity will be detected by the verifying party except with vanishingly small probability.

Non-Computational Protocols

The major drawback with these proofs and other computational techniques is that deep down their security must rely on some computational assumption: the proof of knowledge can be checked if the identifying string is the solution to some hard public problem. If one can solve this problem, he can fake identities. This is the case even if we build the protocol from *perfect* cryptographic tools such as ideal Bit Commitment or ideal Oblivious Transfer.

In the current paper, we consider a situation where two parties, $\mathcal{A}lice$ and $\mathcal{B}ob$, share a common secret string and would like to mutually check their knowledge of that string without disclosing it. This problem has been extensively studied by Fagin, Naor and Winkler [14] who provide a large number of scenarios where the problem may be considered. From the cryptographic point of view only one of their solutions may be considered secure: a solution based on the existence of a one-out-of-two Oblivious Transfer [13] which uses $\Omega(n^2)$ Transfers to do the job for an n-bit secret string.

We describe a simple and efficient protocol based on the exchange of quantum information to check mutual knowledge of a common string in such a way that honest parties will always succeed in convincing each other (except with vanishingly small probability), while a dishonest party interacting with an honest party will have only vanishingly small probability of convincing him. Moreover, a dishonest party gains only a very small amount of information about the secret string from running the protocol: whoever enters the protocol with no knowledge of the secret string would have to enter this protocol an exponential number of times in order to gain non-negligible information about the string.

Our scheme, based on coding theory, depends on no computational assumption, has a total running time of $O(n^{2.376})$ and a total of $O(n^2)$ photons are transmitted (if implemented with a one-out-of-two Oblivious Transfer, only $O(n)$ such Transfers are necessary). We also present a scheme which uses only $O(n)$ photons but that cannot tolerate the transmission errors of a real quantum channel. We suggest the reader consults [8] for more details of quantum cryptography.

We believe such a system should be used in smartcards to avoid frauds from typing PIN codes to dishonest teller machines. A PIN could still be used to activate the functions of the card but it should be typed directly to the card (a device you might as well trust since your bank gives it to you, and they have your money anyway!). The card would identify itself with tellers only through our mechanism: no PIN ever exchanged.

The practical difficulty of our scheme is to embed the necessary technology for the Quantum Oblivious Transfer on a card. Since none of the technology is very fancy we believe such cards could be mass produced (see Section 4.6).

2 Preliminaries

2.1 Notations and Tools

For $b \in \{0,1\}$ we define the selection function $(x,y)_{[b]} = \begin{cases} x \text{ if } b=0 \\ y \text{ if } b=1 \end{cases}$ where x and y are scalars. In general, if x and y are vectors with n components and $\boldsymbol{b} \in \{0,1\}^n$ then $(x,y)_{[\boldsymbol{b}]}$ is the concatenation of $(x_i,y_i)_{[b_i]}$ for all $i \in \{1,2,\ldots,n\}$. If we use a single x instead of a pair then $(x)_{[\boldsymbol{b}]}$ is the concatenation of the x_i's for $b_i = 1$. For instance $(1001)_{[1010]} = 10$. We denote by $\Delta(s,\widehat{s})$ the Hamming distance between s and $\widehat{s}$.

Now let us define what we mean by a secure identification protocol in our context. Suppose $\phi_{(n)} \in_R \{0,1\}^n$ is the secret information shared by $\mathcal{A}lice$ and $\mathcal{B}ob$. In the following we write $\mathcal{B}ob^*$ (resp. $\mathcal{A}lice^*$) to designate somebody trying to impersonate $\mathcal{B}ob$ (resp. $\mathcal{A}lice$).

Definition 1. $\mathcal{B}ob^*$ (or $\mathcal{A}lice^*$) has **almost no information about** ϕ indexed by a security parameter n, $\phi = \{\phi_{(n)} \in_R \{0,1\}^n\}_{n>0}$, if $\mathcal{B}ob^*$'s (or $\mathcal{A}lice^*$'s) information about ϕ can be modeled by a set $\Phi = \{\Phi_{(n)} \subseteq \{0,1\}^n\}_{n>0}$ such that for some $\alpha > 0$ and all sufficiently large n we have:

$$\frac{\#\Phi_{(n)}}{\#\{0,1\}^n} \geq 1 - 2^{-\alpha n} \quad \text{and} \quad \mathrm{P}\left(\phi_{(n)} \notin \Phi_{(n)}\right) < 2^{-\alpha n}$$

and for each $\phi' \in \Phi_{(n)}$ we have:

$$\mathrm{P}\left(\phi_{(n)} = \phi' | \phi_{(n)} \in \Phi_{(n)}\right) = \frac{1}{\#\Phi_{(n)}}.$$

The Shannon information given by Φ about ϕ is such that

$$\mathcal{I}(\phi_{(n)}|\Phi_{(n)}) \leq 2^{-\widehat{\alpha} n}, \widehat{\alpha} > 0.$$

Without loss of generality, we use ϕ for an instance $\phi_{(n)}$ whenever the context permits. We denote by ϕ_i the i^{th} bit of ϕ.

An identification scheme hides the secret information ϕ if when a cheating party runs the protocol with no information about ϕ he ends up with almost no information about ϕ.

Definition 2. An identification scheme **hides the secret information** ϕ shared by $\mathcal{A}lice$ and $\mathcal{B}ob$, if for some $\alpha' > 0$, when $\mathcal{B}ob^*$ (or $\mathcal{A}lice^*$), who starts with no information about ϕ, cheats the protocol $poly(n)$ times, he has probability greater than $1 - 2^{-\alpha' n}$ to end up with almost no information about ϕ.

The random variable ϕ^* denotes the choice made by $\mathcal{A}lice^*$ or $\mathcal{B}ob^*$ to run the protocol given that she (or he) has almost no information about ϕ.

If a malicious party $\mathcal{P}^*$ has almost no information about ϕ then he cannot guess any bit ϕ_i with non-negligible bias.

Property 1 *If $\mathcal{P}^*$ has almost no information about $\phi = \{\phi_{(n)} \in_R \{0,1\}^n\}_{n>0}$ there exists $\beta > 0$ such that for all $\phi^* \in \{0,1\}^n$ and each position i independently, $|P(\phi_i^* = \phi_i) - \frac{1}{2}| \leq 2^{-\beta n}$.*

If $\mathcal{P}^*$ has almost no information about ϕ and executes a protocol leaking almost no information about ϕ then $\mathcal{P}^*$ has still almost no information.

Property 2 *If Φ_0, Φ_1 are two sets that give almost no information about ϕ then $\Phi = \Phi_0 \cap \Phi_1$ gives almost no information about ϕ.*

These two properties are straightforward applications of the above definitions.

2.2 Simple Quantum Transmission

In this paper we consider the most simple idealization of a quantum transmission. There is only four different ways to transmit photons corresponding to the four polarization angles $0°, 45°, 90°, 135°$ that we denote $|\leftrightarrow\rangle, |\nearrow\rangle, |\updownarrow\rangle, |\nwarrow\rangle$ respectively. If $\mathcal{A}lice$ wants to send $b \in \{0,1\}$, she used the following encoding rules:

1. $b = 0$ is randomly encoded by $|\leftrightarrow\rangle$ or $|\nearrow\rangle$.
2. $b = 1$ is randomly encoded by $|\updownarrow\rangle$ or $|\nwarrow\rangle$.

At the receiving end $\mathcal{B}ob$ chooses how he measures the incoming photon either by reading it in rectilinear basis $(|\leftrightarrow\rangle, |\updownarrow\rangle)$, denoted $+$, or in the diagonal basis $(|\nearrow\rangle, |\nwarrow\rangle)$, denoted $\times$. We suppose this is the only choice he can make. If the photon π encodes a bit b in the rectilinear (resp. diagonal) basis and the receiver measures it in the rectilinear (resp. diagonal) basis then he gets the bit b (except if an error occurred during transmission). If π is measured in the diagonal (resp. in the rectilinear) basis then a random bit is received. For a basis $\phi \in \{+, \times\}$ and a bit b we write $\phi_{[b]}$ for the transmission of the encoded bit b in a photon polarized in the basis ϕ. For more details about how quantum transmission works in general consult [8].

3 A basic quantum identification

Suppose $\mathcal{A}lice$ and $\mathcal{B}ob$ who have the secret string ϕ^A and ϕ^B respectively want to test whether $\phi^A = \phi^B$ without revealing their values. In order to achieve this, they are willing to use quantum and public channels. The transmission of $c \in_R \{0,1\}$ polarized in basis $\phi \in_R \{+, \times\}$ hides all information about ϕ to anybody who has almost no a priori information on the values of c and ϕ. This suggests the use of the secret ϕ^A to encode securely the transmission basis.

Suppose $\mathcal{A}lice$ and $\mathcal{B}ob$ share $\phi = \phi^A = \phi^B \in_R \{+, \times\}^n$. In order for $\mathcal{A}lice$ to prove to $\mathcal{B}ob$ that she knows ϕ, she could transmit a random string c taken from a sparse but large subset of $\{0,1\}^n$ polarized in basis ϕ. Therefore, it suffices for $\mathcal{A}lice$ to choose a random codeword c from a code C_n which she sends to $\mathcal{B}ob$. He then measures it in ϕ to obtain the decoded string $\widehat{c}$. If the quantum channel is noiseless then $\widehat{c} = c$. (In the more realistic case, the quantum channel would be modeled as a binary symmetric channel with parameter ϵ.) If there is a large number of codewords in C_n it could be the case that measuring c in basis ϕ^* hides almost all information about ϕ. Conversely if $\mathcal{A}lice$ does not know ϕ it could be very unlikely that she succeeds to send $\widehat{c}$ close to a codeword c, as long as codewords are not too close to each other. Protocol $ident(\phi^A, \phi^B)$, shown below, implements this idea given ϵ the error rates of the quantum channel.

Protocol 3.1 ($ident(\phi^A, \phi^B)$)

1: $\mathcal{A}lice$ and $\mathcal{B}ob$ agree on a binary linear (n, k_n, d_n)-code $C_n \in \mathcal{C}$ by specifying a generating matrix G for C_n.
2: $\mathcal{A}lice$ chooses a random word $x \in_R \{0,1\}^{k_n}$ and takes $c = xG$.
3: $\mathop{DO}\limits_{i=1}^{n}$
- *$\mathcal{A}lice$ sends to $\mathcal{B}ob$ a photon polarized in $\phi^A_{i\,[c_i]}$.*
- *$\mathcal{B}ob$ measures the incoming photon in the basis ϕ^B_i and obtains $\widehat{c}_i$.*

4: $\mathcal{B}ob$ accepts if when decoding $\widehat{c}$, he obtains $c' \in C_n$ such that $\Delta(c', \hat{c}) \leq (\epsilon + \epsilon_0)n$ for $\epsilon_0 > 0$.

Suppose $\mathcal{A}lice$ and $\mathcal{B}ob$ share the same private sequence $\phi = \phi^A = \phi^B$. This implies that, for any $\epsilon_0 > 0$ and except with vanishingly small probability, $\mathcal{B}ob$ will decode $\widehat{c}$ as c which is at Hamming distance less than $(\epsilon + \epsilon_0)n$ from $\hat{c}$ given n sufficiently large.

Now consider a malicious $\mathcal{A}lice$ (denoted by $\mathcal{A}lice^*$) is trying to impersonate the real $\mathcal{A}lice$. We assume that $\mathcal{A}lice^*$ knows almost nothing about ϕ^B at the beginning. Therefore she will have roughly half the positions different from $\mathcal{B}ob$ and thus sends random bits in half the positions. It is easy to show that if the minimum distance d_n of C_n is such that $d_n > 2n(\epsilon + \epsilon_0)$, $\mathcal{A}lice^*$ will be detected with probability greater than $1 - 2^{-\alpha n}$ for $\alpha > 0$. By this attempt, she will not learn more than a vanishingly small amount of Shannon information about ϕ^B.

3.1 $ident(\phi, \phi^*)$ with a dishonest $\mathcal{B}ob$

Let $\phi \in_R \{\boxplus, \boxtimes\}^n$ be $\mathcal{A}lice$'s secret and let c be the transmitted random codeword taken from C_n. $\mathcal{B}ob^*$ chooses a set of bases ϕ^* and measures each photon i in the basis ϕ_i^* in order to obtain $\widehat{c}$. Roughly speaking, one half of the bases of ϕ^* will match with the bases of ϕ. Thus approximatively half of the bits he will receive are the bits of the codeword sent by $\mathcal{A}lice$. The other bits (the positions i for which $\phi_i \neq \phi_i^*$) are not correlated with the bits transmitted. For $\mathcal{B}ob^*$ to be unable to determine a substantial amount of information about ϕ, the code must be chosen so that any half of the bits of the codewords are purely random. Hence, if the proportion of the bits $\mathcal{B}ob^*$ sees about a codeword is random and the rest of the bits he received are not correlated (thus random) the thing he gets is a purely random string. Given that the same would happend for all but a few ϕ^* almost no further information about ϕ can be determined.

A similar concept in a different setting was studied by [4],[24] ((n,j,k)–functions) and [10] (t–resilient functions). The next definition is taken from [4].

Definition 3 [4]. For any integers n, j, k such that $n \geq j + k$, $j > 0$ and $k > 0$, a function $f : GF(Q)^n \rightarrow GF(Q)^j$ is (n,j,k) if, no matter how one fixes any k of its inputs, each of the Q^j outputs can be produced in exactly Q^{n-j-k} different ways by varying the remaining $n - k$ symbols.

If each symbol of $f(x)$ is obtained by computing a weighted sum on a subset of the digits in x then $f(x)$ is said to be xor-(n,j,k). In [4] the function f is from n–bit strings to j–bit strings, here we consider an arbitrary field. The following theorem showing how to construct (n,j,k)–functions, was originally proved for functions over binary strings. It is straightforward to generalize it to functions over arbitrary fields.

***Theorem 4** [4]. For a set of values (n,j,k), there exists an xor-(n,j,k)-function from $GF(Q)^n$ to $GF(Q)^j$ if and only if there exists an $[n,j,k+1]$ linear codes C_n over $GF(Q)$.*

If G is the generating matrix of a $[n,j,k+1]$-code C_n then the function $f(x) = Gx^T$ (x^T is x transposed) is $xor - (n,j,k)$. Saying that f is (n,j,k) implies that $f^{-1}(x)$ is a set which have uniform projection on any k coordinates [10]:

Definition 5 [10]. A set $S \subseteq GF(Q)^n$ has a uniform projection on any j components if for all $\omega \in \{0,1\}^n$ of weight j and all $a \in GF(Q)^k$, the set $S_{\omega,a} = \{x \in S : (\perp, x)_{[\omega]} = a\}$ is such that $\#S_{\omega,a} = \frac{\#S}{Q^k}$.

If $f(x) = Gx^T$ is (n,j,k) then $f^{-1}(x) = Hx^T$, where H is the parity check matrix for C_n. The matrix H is also the generating matrix for the dual $C_n^\perp$ of C_n. The next theorem makes the connection between uniform projections and some conditions on the dual of the codes C_n.

Theorem 6. *If there exists a family of codes $\mathcal{C} = \{C_n\}_{n>0}$ such that for n sufficiently large the dual $C_n^\perp$ of C_n has minimum distance d'_n with $\frac{d'_n}{n} \geq \zeta > \frac{1}{2}$, protocol ident$(\phi^A, \phi^*)$ hides almost all information about ϕ^A to $\mathcal{B}ob^*$ except with probability exponentially small in n. This holds given an idealized quantum transmission.*

proof sketch: By property 1, except with vanishingly small probability $\Delta(\phi^*, \phi^A) \geq (1-\zeta)n$. Thus, the number of positions for which $\mathcal{B}ob^*$ sees the bits of c is less than ζn. Let H be the generating matrix for $C_n^\perp$ of minimum distance ζn. The function Hx^T has uniform projection on any ζn components. Thus, any ζn bits of a codeword c (those for which $Hc^T = \mathbf{0}$) are random when $c \in_R C_n$. Since the other $(1-\zeta)n$ positions are random, the string $\hat{c}$ he received is purely random. Therefore, the set $\Phi_1 = \{\phi | \Delta(\phi, \phi^*) \geq (1-\zeta)n\}$ models the knowledge leaked to $\mathcal{B}ob^*$ by the actual execution of the protocol. It is easy to show that Φ_1 gives almost no information about ϕ. Let Φ_0 be the model for $\mathcal{B}ob^*$'s knowledge about ϕ^A before the actual execution. The set $\Phi = \Phi_0 \cap \Phi_1$ models $\mathcal{B}ob^*$'s knowledge after the current execution given he had almost no information when entering the protocol. By property 2 the set Φ gives almost no information about ϕ^A. For $\mathcal{B}ob^*$ executing $poly(n)$ times the protocol, the set

$$\Phi = \bigcap_{i=1}^{poly(n)} \Phi_i$$

where Φ_i models the information leaked about ϕ for the i^{th} execution gives almost no information about ϕ^A. We conclude that $ident(\phi^A, \phi^*)$ hides almost all information about ϕ^A to $\mathcal{B}ob^*$. $\square$

3.2 Code Properties

Over the last few sections we have suggested some conditions on our codes. Let us now summarize the properties that families of codes $\mathcal{C}$ must satisfy to guarantee the security of protocol $ident(\phi^A, \phi^B)$ while preserving efficiency of the scheme.

1. Given our mode of transmission via the quantum channel, we want $\mathcal{C}$ to be a family of binary codes.
2. Each $C_n \in \mathcal{C}$ must be efficiently constructible and efficiently decodable.
3. Each $C_n \in \mathcal{C}$ must have minimal distance d_n such that $\frac{d_n}{n} \geq 2(\epsilon + \epsilon_0)$ for $\epsilon_0 > 0$.
4. The dual $C_n^\perp$ of $C_n \in \mathcal{C}$ must have minimal distance d'_n such that $\frac{d'_n}{n} \geq \zeta$ for $\zeta > \frac{1}{2}$.

The set of conditions above is bad news. First of all, these conditions cannot be satisfied because of Plotkin's bound on codes [20] when $\zeta > \frac{1}{2}$. Fortunately, modifications to the protocol (for instance by using more than two transmission bases) open the possibility of relaxing this condition to $\zeta > 0$. But even then, no known family of codes satisfies these four conditions at once. It is nevertheless easy to find codes meeting any three of them. For instance, concatenated codes [16] achieve conditions 1, 2 and 3, random binary linear codes meet conditions 1,3 and 4, while Reed-Solomon codes meet 2,3 and 4.

It is common in coding theory to take care of arbitrary long messages via block codes. These codes are of no help in our setting because their duals have small minimum distance. This is easy to see since it is sufficient to observe a constant number of bits to tell if a word is a candidate codeword or not.

On top of these problems due to coding theory, more fundamental problems arise from our protocol: we have made a very strong assumption that $\mathcal{A}lice^*$ and $\mathcal{B}ob^*$ send and receive photons in only two possible bases. In reality we would have to deal with the fact that they can use very different quantum states and quantum measurements. It is indeed completely unknown to us if this protocol is safe under these general conditions.

The main problem in quantum cryptography is to provide proofs for the security of cryptographic primitives assuming the most general quantum measurements an opponent could make. Nevertheless, the full security of the quantum bit commitment primitive has been obtained in [5] and quantum oblivious transfer has been shown secure against a large set of measurements[21]. Basing our identification scheme on such primitives gives more freedom on the codes while, at the same price, providing security against any quantum measurements. Oblivious transfer has already been used by [14] to solve the problem of identification. In the next section, we present a different solution based on quantum oblivious transfer and theorem 6.

4 The Final Protocol

No existing family of codes meets the four conditions above. One way around this problem is to drop condition 1. To do so we need a means of transferring elements of a larger field $GF(Q)$ at once. This is exactly the purpose of a $\binom{2}{1}$–OST, a *One-Out-of-Two Oblivious String Transfer* [11]. We can thus modify our protocol to use this primitive instead of the quantum transmission of section 3. A nice side effect of this modification is that transmission errors also go away.

Doing this modification is not very costly since a $\binom{2}{1}$–OST can be implemented using a constant number of $\binom{2}{1}$–OT_2 [11], which in turn can be obtained from the quantum transmission [6]. The solution we describe next works over $GF(4)$ (and thus we use a $\binom{2}{1}$–OT_4).

The Appendix provides a modified protocol (from [6]) for Quantum Oblivious Transfer ($\binom{2}{1}$–OT_4) sufficient for this application. That protocol also relies on the existence of a bit commitment. To avoid computational assumptions again at that level we recommend using the Quantum Bit Commitment Scheme of [5]. The protocol of the next subsection combined with the one from the Appendix constitute the complete Quantum Oblivious Mutual Identification.

4.1 Protocol

Suppose $\mathcal{A}lice$ and $\mathcal{B}ob$ share $\phi = \phi^A = \phi^B \in_R \{0,1\}^n$. In order for $\mathcal{A}lice$ to prove to $\mathcal{B}ob$ that she knows ϕ, she transmits a random string c taken from a sparse but large set C, in such a way that if they agree on the same ϕ he receives c and verifies that it belongs to C, but otherwise receives a rather random string which is unlikely to be in C.

More precisely, let M be a random $n \times k$ matrix over $GF(4)$. (In banking applications, M is chosen by the bank and may be made public.) Let C_n be the $[n, k_n, d_n]$ linear code over $GF(4)$ generated by M. Let $C_n^{\perp}$ be the $[n, n-k_n, d'_n]$ linear code over $GF(4)$ dual to C_n.

Protocol 4.1 (*identOT*$(\phi^A)(\phi^B)$)

1: $\mathcal{A}lice$ *picks* $r, s \in_R \{00, 01, 10, 11\}^n$.
2: $\mathbf{DO}_{i=1}^{n}$ $\mathcal{A}lice$ *runs* $\binom{2}{1}\text{–}OT_4(r_i, s_i)(\phi_i^B)$ *with* $\mathcal{B}ob$ *who receives* v_i.
3: $\mathcal{B}ob$ *picks* $x, y \in_R \{00, 01, 10, 11\}^n$ *and announces it to* $\mathcal{A}lice$.
4: $\mathcal{A}lice$ *picks* $c \in_R C_n$, *computes and sends* $u \leftarrow c \oplus (r \oplus x, s \oplus y)_{[\phi^A]}$.
5: $\mathcal{B}ob$ *accepts only if* $u \oplus v \oplus (x,y)_{[\phi^B]} \in C_n$.

In the above protocol, in contrast to protocol 3.1, the randomization of step 3 is necessary because the $\binom{2}{1}$–OT_4 no longer provides the fact that a random element is obtained when $\phi_i^* \neq \phi_i^B$, and we do not want $\mathcal{A}lice$ to take advantage of that fact. From the previous sections it is now easy to see why the protocol is correct and secure.

4.2 *identOT*(ϕ, ϕ) with honest parties

If $\mathcal{A}lice$ and $\mathcal{B}ob$ share the same private sequence $\phi = \phi^A = \phi^B$ then

$$u \oplus v \oplus (x,y)_{[\phi]} = c \oplus (r \oplus x, s \oplus y)_{[\phi]} \oplus (r,s)_{[\phi]} \oplus (x,y)_{[\phi]} = c.$$

Therefore $\mathcal{B}ob$ accepts.

4.3 *identOT*(ϕ^*, ϕ^B) with a dishonest $\mathcal{A}lice$

Suppose a malicious $\mathcal{A}lice$ (denoted by $\mathcal{A}lice^*$) tries to impersonate the real $\mathcal{A}lice$. We assume that $\mathcal{A}lice^*$ knows almost nothing about ϕ^B at the beginning. This section specifies code parameters that allow $\mathcal{B}ob$ to reject $\mathcal{A}lice^*$ with probability exponentially close to one.

Now we show how to choose the parameter k of code in order to eliminate the chances that $\mathcal{A}lice^*$ identify as $\mathcal{A}lice$ successfully.

Theorem 7. *If* $d_n \in \Omega(n)$ $\mathcal{B}ob$ *will reject* $\mathcal{A}lice^*$ *except with probability exponentially small in* n.

proof sketch: $\mathcal{B}ob$'s final calculation is $u \oplus v \oplus (x,y)_{[\phi^B]}$ which by definition of v is $u \oplus (r \oplus x, s \oplus y)_{[\phi^B]}$. $\mathcal{A}lice^*$, who knows r,s,x,y may try to choose a u cleverly to make this a codeword for as many ϕ^B as possible. We show this is not possible.

First notice that for any fixed r,s, when x,y are uniformly chosen at random $\Delta(r \oplus x, s \oplus y) \approx 3n/4$. By the assumption that $\mathcal{A}lice^*$ has almost no information about ϕ^B we know that the number of equally likely candidates for ϕ^B is roughly 2^n. Fix a u and a ϕ^B and assume the corresponding word has syndrome S. We know that all the ϕ' that are different from ϕ^B in the positions where $r \oplus x$ and $s \oplus y$ are the same will not change the result of the calculation. Therefore roughly $2^{n/4}$ values of ϕ^B will yield the same result with syndrome S.

On the other hand, any ϕ' that differ from ϕ^B in positions where $r \oplus x$ and $s \oplus y$ are different will yield a different result. As long as the number of differences is no more than $d_n/2$ the resulting words cannot have the same syndrome because this would imply that the code contains a word of weight less than d_n and no other word of syndrome S can be closer than the one we started from. Therefore $2^{n/4}\sum_{i=1}^{d_n/2}\binom{3n/4}{i}$ possibilities for ϕ^B will yield results of another syndrome and each of these is associated to a single word of syndrome S.

In conclusion, for any fixed u and S, there is always at least $\sum_{i=1}^{d_n/2}\binom{3n/4}{i}$ times more possibilities for ϕ^B that do not yield a word of syndrome S than those that yield a word of syndrome S. If $d_n \geq 2\delta n$ for some constant δ, this value is roughly $2^{\frac{3H(\delta)}{4}n}$. Therefore the probability that $\mathcal{A}lice^*$ gets $\mathcal{B}ob$ to accept is no more than $2^{-\alpha n}$ with $\alpha = 3H(\delta)/4$.

□

We need to know more than just the fact that $\mathcal{A}lice^*$ will fail most of the time: we show that in the case of failure she cannot learn much about ϕ_B.

Theorem 8. *If $d_n \in \Omega(n)$ $\mathcal{A}lice^*$ learns almost nothing about ϕ_B, except with probability exponentially small in n.*

proof sketch: By the same counting argument as above, there cannot be more than $2^{(1-3h(\epsilon)/4)n}$ possibilities of ϕ_B that would yield codewords. When rejected, $\mathcal{A}lice^*$'s only gain in knowledge is that the real ϕ_B was not one of those. Thus she can eliminate only that many strings. □

The consequence of these theorems is that with probability $1-2^{-\alpha n}$ $\mathcal{A}lice$ will be rejected and in that case she may discard only $2^{(1-3h(\epsilon)/4)n}$ strings as candidates for ϕ^B. She will thus still have almost no information about ϕ^B even after discarding $poly(n) \times 2^{(1-3h(\epsilon)/4)n}$ strings.

4.4 $identOT(\phi^A, \phi^*)$ with a dishonest $\mathcal{B}ob$

Now we analyze the situation from the point of view of an honest $\mathcal{A}lice$ facing a malicious $\mathcal{B}ob^*$.

Theorem 9. *Except with exponentially small probability in n, Protocol $identOT(\phi)$ hides all information about ϕ^A to $\mathcal{B}ob^*$ if the code is a $[n,k_n,d_n]$-code with a dual $[n, n-k_n, d'_n]$-code such that $d'_n \geq (\frac{1}{2}+\gamma)n$ for $0<\gamma<\frac{1}{2}$.*

proof sketch: For each position i such that $\phi_i^* = \phi_i^A$, Bob will get the codeword position c_i of c. The $\#\{i|\phi_i^A = \phi_i^*\}$ is at most $(\frac{1}{2}+\gamma)n$ except with probability smaller than $2e^{-\gamma^2 n}$. The remaining positions (j) of $(x,y)_{[\phi^B]} \oplus u \oplus v$ for $\phi_j^* \neq \phi_j^A$ are not correlated with c_j as long as r and s were originally chosen at random. Since the dual has minimum distance $d' > (\frac{1}{2}+\gamma)n$ we conclude that theorem 6 applies and that Protocol *IdentOT* hides the information ϕ^A to $\mathcal{B}ob^*$. □

Example: Codes with such properties exist over $GF(4)$. For instance, a random $n \times 0.91n$ 4-ary matrix is likely to define a $[n, 0.91n, 0.02n]$ code with a $[n, 0.09n, 0.52n]$ dual code. Assymptoticaly, the probability that such a matrix do not define a code with these parameters is exponentially small in n.

4.5 Complexity

Protocol 4.1 runs in time $O(n^2)$ (to choose a codeword) and uses $O(n)$ $\binom{2}{1}$–OT_4, where n is the security parameter. When combined with the sub-protocol from the Appendix the total running time of the final protocol becomes $O(n^3)$ and a total of $O(n^2)$ photons are transmitted. The running time decreases to $O(n^{2.376})$ if more efficient codes such as the Superconcentrator Codes of Spielman [25] are used in the protocol of the Appendix in both the Oblivious Transfer and the Bit Commitment (time $O(n^2)$) and if all the commitments are done at once in order to save on the time necessary to compute the hash function ($O(n)$ products of a vector by a matrix with a total of $O(n^3)$ operations may be replaced by a matrix product which takes only time $O(n^{2.376})$).

We must point out that despite the fact that our Protocol 4.1 is more efficient than that of Fagin, Naor and Winkler [14] in terms of $\binom{2}{1}$–OT_2, when used together with the Quantum Oblivious Transfer their protocol can be made as efficient as ours in terms of photons. This is because their protocol requires n transfers of n-bit strings (which implies $\Omega(n^2)$ $\binom{2}{1}$–OT_2) while our protocol requires only n transfers of 2-bit strings (which can be done with $O(n)$ $\binom{2}{1}$–OT_2). The fact is that the Quantum Oblivious Transfer can be used to transfer 1, 2 or up to $O(n)$ bits at no extra cost. In order to have a real gain in terms of photons transmitted we need a Quantum O-T that requires only a constant number of photons to transfer a constant number of bits (see open problem 3).

4.6 Implementation Remarks

The protocol from the appendix uses quantum transmission both for Oblivious Transfer and Bit Commitment. At first glance, it seems like the quantum transmission of data must go in both directions, since the Oblivious Transfer goes from $\mathcal{A}lice$ to $\mathcal{B}ob$ and the Bit Commitment goes the other way. As pointed out in [12], there is no need for photons traveling both ways. These two protocols may be implemented with the photons going in a single direction. It does not matter who send the photons to who, the same result can be achieved from them. (A similar idea was suggested by Hans-Joachim Knobloch [19].)

Because of the above remark, in a smartcard scenario it suffices to implement on the card the technology for sending polarized photons: a weak light source with a multiple polarizer system. As for the ATM it would have to use the more elaborate technology for making polarization measurements on the incoming photons. Since the distance between the sender and receiver could be of a few millimeters the actual error rate of the quantum transmission would be extremely low (error rates of 0.5% have been observed on hundreds of meters [22]).

5 Conclusion and Open Questions

We have presented a protocol for mutual identification based on the existence of an Oblivious Transfer and have shown improvements to the Quantum Oblivious Transfer in order to combine them in an efficient Quantum Mutual Identification Protocol. Here is a few open problems:

1. It would be interesting to show that the protocol of section 3 is secure even if the participants use arbitrarily complicated physics.
2. Find binary codes satisfying conditions 2,3, and 4.
3. Find a reduction of $\binom{2}{1}$–OST to the quantum transmission that requires only a constant number of photons per word of constant length.
4. Implement the necessary technology on a smartcard!

Acknowledgments

We would like to thank Thomas Beth, Gilles Brassard, Artur Ekert, Richard Jozsa, Denis Langlois, and Bill Wootters for their interest in this research.

References

1. Anderson, R.J., "Why Cryptosystems Fail", in *Proceedings of the 1993 ACM Conference in Computer and Communications Security* pp 215 - 227
2. Ash, R., *Information Theory*, John Wiley & Sons, 1965.
3. Bennett, C.H., G. Brassard, *Quantum Cryptography: Public key distribution and coin tossing*, Proc. of IEEE International Conference on Computers, Systems, and Signal Processing, Banglore, India, December 1984, pp. 175–179.
4. Bennett, C.H., G. Brassard, J.-M. Robert, *Privacy Amplification by Public Discussion*, SIAM Journal on Computing, Vol. 17, No.2, 1988, pp. 210–229.
5. Brassard, G., C. Crépeau, R. Jozsa, D. Langlois, *A quantum bit commitment scheme provably unbreakable by both parties*, Proceeding of the 34th annual IEEE Symposium on Foundations of Computer Science, November 1993,pp. 362–371.
6. Bennett, C.H., G. Brassard, C. Crépeau, M.-H. Skubiszewska, *Practical Quantum Oblivious Transfer*, In proceedings of CRYPTO'91, Lecture Notes in Computer Science, vol 576, Springer Verlag, Berlin, 1992, pp 351–366.

7. Brassard, G., *On computationally secure authentication tags requiring short secret shared keys*, Advances in Cryptology: Proceedings of CRYPTO 82, Plenum Press, 1983, pp.79–86.
8. Brassard, G., *Cryptology column — Quantum cryptography: A bibliography*, Sigact News, vol. 24, no. 3, 1993, pp.16–20.
9. Carter, J.L., M. N. Wegman, *New Hash Functions and Their Use in Authentication and Set Equality*, Journal of Computer and System Sciences, Vol. 22, 1981, pp. 265–279.
10. Chor, B., O. Goldreich, J. Hastad, J. Freidman, S. Rudich, R. Smolensky, *The bit extraction problem or t–resilient functions*, Proc. 26th IEEE Symposium on Foundation of Computer Science, Portland, Oregon, 1985, pp.396–407.
11. Crépeau, C. and M. Sántha. *Efficient reductions among oblivious transfer protocols based on new self-intersecting codes.* In Sequences II, Methods in Communications, Security, and Computer Science, pp. 360–368. Springer-Verlag, 1991.
12. Crépeau, C., *Quantum Oblivious Transfer*, Journal of Modern Optics, Dec. 1994.
13. Even, S., Goldreich, O. and Lempel, A., "A randomized protocol for signing contracts", *Communications of the ACM*, vol. 28, 1985, pp.637–647.
14. Fagin, R., M. Naor and P. Winkler, *Comparing Common Secret Information without Leaking it*, submitted for publication, Communications of the ACM, 1994.
15. Fiat, A and A. Shamir. *How to prove yourself: practical solutions to identification and signature problems.* In A. M. Odlyzko, editor, Proceedings CRYPTO 86, pages 186–194. Springer, 1987. Lecture Notes in Computer Science No. 263.
16. Forney, G. D., *Concatenated Codes*, The M.I.T. Press, 1966.
17. Goldreich, O., S. Goldwasser, and S. Micali. *How to construct random functions.* In Proceedings of the 25th IEEE Symposium on Foundations of Computer Science, pp. 464–479, Singer Island, 1984. IEEE.
18. Kilian, J., *Founding cryptography on oblivious transfer.* In Proc. 20th ACM Symposium on Theory of Computing, pp. 20–31, Chicago, 1988. ACM.
19. Knobloch, H.-J.,*personal communication* through T. Beth.
20. Mac Williams, F.J. and N.J.A. Sloane, *The Theory of Error-Correcting Codes*, North-Holland, 1977.
21. Mayers, D., L. Salvail, *Quantum Oblivious Transfer is Secure Against Individual Measurements*, In the Proceedings of PHYSCOMP 94, Dallas, 1994, pp. 69–77.
22. Muller, A., Breguet, J. and Gisin, N., *Experimental demonstration of quantum cryptography using polarized photons in optical fiber over more than 1 km* In Europhysics Letters, vol. 23, no. 6, 20 August 1993, pp..383–388.
23. Rabin, M. O., *How to exchange secrets by oblivious transfer*, Technical Memo TR–81, Aiken Computation Laboratory, Harvard University, 1981.
24. Robert, J.-M., *Détection et correction d'erreur en cryptographie*, Master thesis, Département d'informatique et de Recherche Opérationnelle, Université de Montréal, Montréal, Québec, Canada, 1985.
25. Spielman, D., *Linear-time Codable and Decodable Error-Correcting Codes.* In Proc. 27th ACM Symposium on Theory of Computing, 1995. ACM.
26. *One Less Thing to Believe In: Fraud at Fake Cash Machine*, New York Times, 13 May 1993, pp. A1 & B9.

Appendix

We refer the reader to [6, 5, 12] for more details on this protocol. Here ϵn is a limit on the number of errors that can be tolerated from real noise. The actual error rate ϵ' should be less than ϵ in order to reject an honest *Alice* accidentally only with probability exponentially small in n.

Protocol 5.1 ($\binom{2}{1}$–$\mathbf{OT}_4(q_0, q_1)(c)$)

1: $\underset{i=1}{\overset{n}{\textbf{\textit{DO}}}}$
- *Alice picks* $r_i \in_R \{0.1\}$ *and* $\beta_i \in_R \{+, \times\}$,
- *Bob picks* $\beta'_i \in_R \{+, \times\}$,
- *Alice sends to Bob a photon* π_i *with polarization* $\beta_{i[r_i]}$,
- *Bob measures photon* π_i *in basis* β'_i *and obtains a bit* r'_i.

2: *Bob runs commit*$(r'_1 r'_2 ... r'_n \beta'_1 \beta'_2 ... \beta'_n)$ *with Alice.*

3: *Alice picks a random bit* t *and announces it to Bob.*

4: ***if*** $t = 0$ ***then***
- *Bob runs unveil*$(r'_1 r'_2 ... r'_n \beta'_1 \beta'_2 ... \beta'_n)$,
- *Alice checks that* $\#\{i \mid \beta_i = \beta'_i \text{ and } r_i \neq r'_i\} < \epsilon n$,
- *Alice and Bob* ***restart the protocol.***

5: *Alice announces her choices* $\beta_1 \beta_2 ... \beta_n$ *to Bob.*

6: *Bob randomly selects two subsets* $I_0, I_1 \subset \{1, ..., n\}$ *subject to* $|I_0| = |I_1| = n/2$, $I_0 \bigcap I_1 = \emptyset$ *and* $\forall i \in I_0, \beta_i = \beta'_i$ *or* $\forall i \in I_1, \beta_i \neq \beta'_i$, *and announces* $I_c, I_{\bar{c}}$ *to Alice.*

7: *Alice*
- *receives* J_0, J_1, *and defines* $w_0 \leftarrow r_{j_1^0} r_{j_2^0} ... r_{j_{n/2}^0}$ *and* $w_1 \leftarrow r_{j_1^1} r_{j_2^1} ... r_{j_{n/2}^1}$ *with* $j_l^b \in J_b$ *and* $j_l^b < j_{l+1}^b$ *for* $b \in \{0, 1\}$ *and* $1 \leq l < n/2$,
- *computes their syndromes* $S_0 \leftarrow Syn(w_0)$ *and* $S_1 \leftarrow Syn(w_1)$ *with respect to an agreed upon linear code* C *(consult [6] for details),*
- *picks a random privacy amplification function* $h : \{0, 1\}^{n/2} \rightarrow \{0, 1\}^2$,
- *computes* $\hat{q}_0 \leftarrow q_0 \oplus h(w_0)$ *and* $\hat{q}_1 \leftarrow q_1 \oplus h(w_1)$,
- *sends* $S_0, S_1, h, \hat{q}_0, \hat{q}_1$ *to Bob.*

8: *Bob*
- *receives* $S_0, S_1, h, \hat{q}_0, \hat{q}_1$,
- *computes a word* $\hat{w}$ *of syndrome* S_c *using the decoding algorithm of* C *from word* $w' = r'_{i_1} r'_{i_2} ... r'_{i_{n/2}}$ *with* $i_l \in I_0$ *and* $i_l < i_{l+1}$ *for* $1 \leq l < n/2$,
- *computes and returns* $q_c \leftarrow \hat{q}_c \oplus h(\hat{w})$.

The privacy amplification function used in Step 7 can be a the concatenation of the XOR of two random subsets of the bits of its input. In the Bit Commitment protocol of Step 2, a single privacy amplification function can be used for all of them.

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