A method of Authentication for Quantum Networks

Stefan Rass

***Abstract*— Quantum cryptography offers a way of key agreement, which is unbreakable by any external adversary. Authentication is of crucial importance, as perfect secrecy is worthless if the identity of the addressee cannot be ensured before sending important information. Message authentication has been studied thoroughly, but no approach seems to be able to explicitly counter meet-in-the-middle impersonation attacks. The goal of this paper is the development of an authentication scheme being resistant against active adversaries controlling the communication channel. The scheme is built on top of a key-establishment protocol and is unconditionally secure if built upon quantum cryptographic key exchange. In general, the security is the same as for the key-agreement protocol lying underneath.**

***Keywords*— meet-in-the-middle attack, quantum key distribution, quantum networks, unconditionally secure authentication.**

I. Introduction

AUTHENTICATION is a crucial aspect for many applications in the area of cryptography. Quantum key distribution (QKD) offers unconditionally secure message transfer, but cannot ensure the identity of the communication partner on the other side. So, although the messages are perfectly concealed, an adversary sitting on the other end of the line will get the message if s/he can successfully impersonate the true message addressee. Commonly used authentication mechanisms usually follow one of three paradigms: Password-Authentication, challenge-and-response or zero-knowledge. All of them can be broken by meet-in-the-middle attacks. Moreover, some challenge-and-response techniques as well as zero-knowledge proofs of identity rely on intractability assumptions, hence are not unconditionally secure. The goal of this paper is the development of a method for unconditionally secure authentication for *quantum networks*, i.e. networks in which adjacent nodes are able to exchange secrets by means of quantum cryptography. We will assume that key distribution is feasible, i.e. adjacent nodes can efficiently exchanges secret keys of arbitrary length. Under this assumption, we will provide an authentication scheme that can effectively counter a meet-in-the-middle attacks. We allow an adversary to act arbitrarily, that is s/he may relay, replace, block or insert new messages from Alice to Bob and vice versa.

The approach has some valuable properties: Firstly, it is purely computational, thus does not require any hardware purely dedicated to the authentication protocol. Secondly, the concept is not limited to any particular OSI layer, hence can be embedded where most appropriate or efficient. Finally, it is highly efficient, as we only require few operations and little data to be transmitted.

Our scheme can be generalized, as quantum cryptographic key-agreement can be replaced by an arbitrary key-agreement protocol. The resulting protocol is as secure as the underlying key-agreement scheme is.

Authentication techniques relying on similar ideas as quantum cryptography does, i.e. transmitting information encoded within photons, cannot benefit from the eavesdropping detection feature as normal quantum cryptography does, because we consider the adversary sitting on the other end of the communication channel. We shall provide an authentication method *not* relying on any quantum encoding nor relying on quantum computers or on any constraint on the noise on the channel.

Although there is a lot of literature available on authentication ([17], [10], [19], [15], [11] to mention just a few), the author is not aware of any approach explicitly dealing with the avoidance of meet-in-the-middle attacks, as most schemes focus on authentication of messages or parties in a setup where the adversary is just passively listening (especially the zero-knowledge paradigm aims at preventing information-leakage [7], [4], [12]). To the best of the authors knowledge, this is the first attempt to exploit the properties of quantum key distribution in order to detect an impersonation attack.

The remainder of this article is organized as follows: Section II describes the context where our protocol can be implemented to be beneficial. In section III, we describe the adversary model, and provide the authentication scheme together with formal results showing its security. Section IV gives hints on variations of the protocol. Concluding remarks are given in section V.

We will not explicitly go into details about mutual authentication, as this can always be achieved by executing simple one-way authentication in both directions. Therefore, we restrict ourselves to the uni-directional case.

II. Context

Assume a network where adjacent nodes are capable of efficient quantum key distribution [2], [1], i.e. shared secrets being uniformly distributed and arbitrarily long can be established between two nodes. As nowadays quantum cryptography is limited to point-to-point connections, several attempts for an integration of QKD within protocols of higher layers have been published (see [5], [6]).

A useful model of key exchange is the following: Alice and Bob possess two random variables X and Y being correlated. The eavesdropper can sample from a third random variable Z, being correlated to X and Y but less correlated as X and Y are. Key distillation protocols provide techniques letting Alice and Bob sample from their variables X, Y and extract a key, of which Eve has negligibly little information about. If we consider the communication between Alice and Bob fully running through an intermediate node being under Eve's

University of Klagenfurt, System Security Research Group, Transportation Informatics Group, Austria (email: stefan.rass@uni-klu.ac.at)

control, then Eve may intercept the message flow, such that (X, Z) and (Y, Z) enjoy stronger correlation than (X, Y). In that case, key-establishment fails to be secret. Hence, we need to cut down the amount of information Eve is getting by listening on the channel.

A straightforward solution is the usage of multiple paths between Alice and Bob, so Eve has to intercept more than one path in order to successfully mount an attack. Graph-theory can be applied to build such networks and suitable protocols have been designed [13]. However, to avoid a situation as above, where all traffic passes a single (possibly adversarial) node, we need to ensure that the adversary cannot impersonate other nodes. More abstractly, we wish to repel the classical meet-in-the-middle attack, where the adversary can intercept and modify all traffic between Alice and Bob.

Unconditionally secure authentication techniques can be implemented using hash-functions of different nature, the most renowned of which is the Wegman-Carter MAC [20], which we will utilize in this report. Our solution uses message authentication to prove ones identity. Other hash-function based possibilities, which are not considered here, include bucket-hashing [15], and evaluation hash functions [11].

III. Repelling the Meet-in-the-Middle Attack

Consider the following setup depicted in figure 1: To initiate a communication, Alice exchanges a key σ with a remote node X, which she believes to be Bob. Bob does the same, i.e. exchanges a key $\overline{\sigma}$ with a remote node Y, which pretends to be Alice. In other words, with no adversary in the middle, we have $X = \text{Bob}, Y = \text{Alice}$ and $\sigma = \overline{\sigma}$. However, if Eve sits in the middle, then we have $X = Y = \text{Eve}$, so Eve shares secrets σ and $\overline{\sigma}$ with Alice and Bob. Eve can passively relay the messages from Alice to Bob and vice versa. Regardless of the protocol Alice and Bob use for authentication, Eve will not be detected. Certainly, if approaches similar to quantum cryptographic key-exchange are used, then Eve can be detected while listening on the communication channel, however, we do not benefit from the eavesdropping-detection feature of quantum techniques, as Eve just has to *wait* until Alice and Bob have successfully authenticated each other. As soon as mutual confidence is established, Eve can read all traffic, by intercepting and re-sending each message appropriately. Note that under this setup, Eve will be successful *regardless* of the authentication protocol, which Alice and Bob execute. This section shall provide a method for avoiding this attack, hence realizing an unconditionally secure authentication of Alice and Bob even in the presence of an adversary sitting in the middle.

Based on the assumption that $\sigma \neq \overline{\sigma}$ (see section III-D), detection of Eve intuitively proceeds as follows: Assume that prior to any communication, Alice and Bob possess a common secret r, which is completely unknown to Eve (exchanged by non-cryptographic means for instance). Alice and Bob possess keys σ and $\overline{\sigma}$ being at least twice as long as r, say n bit, i.e. $2n = |\sigma| = |\overline{\sigma}| = 2|r|$. The parameter n is publicly known to all parties. [1] For Alice and Bob to authenticate each other, Alice takes the first n bits k of σ, attaches a suitable MAC $z = s(k, r)$ using her private shared secret r and sends the message (k, z) to Bob. Two cases can be distinguished:

1) There is no adversary in the middle. Then Bob successfully verifies $\widetilde{z} = \overline{z}$, where $\widetilde{z}$ is the value he received from "Alice" and $\overline{z}$ is the signature he calculated from the first n bit of his key $\overline{\sigma}$. If this verification succeeds, Bob takes the second n-bit block from $\overline{\sigma}$ and uses them for proving his identity to Alice.
2) Eve sits in the middle and possesses secrets shared with Alice and Bob. In this case, Eve should not be able to choose the bits of σ and $\overline{\sigma}$ freely, so we will have $\sigma \neq \overline{\sigma}$. Forging the message (k, z) such that $(\overline{k}, \overline{z})$ is received and accepted by Bob, should not work because of the signature z which is dependent on r, which in turn is unknown to Eve. Hence, for an unconditionally secure "signature" z, Eve will always be detected.

We will give hints on suitable signature functions in the next section. It remains to prevent Eve from biasing σ and $\overline{\sigma}$ such that $\sigma = \overline{\sigma}$. However, this can be accomplished easily, as we will show in section III-D.

A. Authentication Codes

As previously mentioned, hash-functions can be used to construct authentication codes which are unconditionally secure.

Let $\mathcal{K}$ denote the key-space. If m is a message, $k \in \mathcal{K}$ is a secret shared by the sender and receiver and s is a (signature) function, then the information $z = s(m, k)$ is attached to m and sent to the receiver. This one in turn calculates $\widetilde{z} = s(\widetilde{m}, k)$ from the $\widetilde{m}$ he received and verifies whether $z \stackrel{?}{=} \widetilde{z}$. If this holds, then the message is believed to be authentic. An active adversary has two options: S/he may introduce a new message (m, z) thus pretending to be a particular sender. This is called an *impersonation attack*, for obvious reasons. Alternatively, Eve can observe the message (m, z) and replace it with another message $(\widetilde{m}, \widetilde{z})$ with $m \neq \widetilde{m}$, which is called a *substitution attack*. The success probability is denoted as P_S or P_I for *substitution* or *impersonation*, respectively. The probabilities can be expressed as $P_I = \max_{m,z} \Pr((m, z) \text{ is valid})$ and $P_S = \max_{m,z} \max_{\widetilde{m} \neq m, \widetilde{z}} \Pr((\widetilde{m}, \widetilde{z}) \text{ is valid} | (m, z) \text{ is observed})$. Assuming that the authentication keys k are uniformly distributed on $\mathcal{K}$, we can write these probabilities as follows ([8]):

$$P_I = \max_{m,z} \frac{|\{k \in \mathcal{K} | z = s(m,k)\}|}{|\{k \in \mathcal{K}\}|}$$
$$P_S = \max_{m,z} \max_{\widetilde{m} \neq m, \widetilde{z}} \frac{|\{k \in \mathcal{K} | z = s(m,k), \widetilde{z} = s(\widetilde{m},k)\}|}{|\{k \in \mathcal{K} | z = s(m,k)\}|}$$

In the following, we will define $h_k(\cdot) := s(\cdot, k)$. The key k will implicitly be present as a parameter for the selection of a particular h from the set of hash-functions $\mathcal{H} = \{h_k : A \to B | k \in \mathcal{K}\}$. For our purposes, two classes of hash-functions will suffice (cf. [19], [18]).

Definition 3.1: A family $\mathcal{H}$ of hash-functions is called ε-*almost universal*$_2$ if for any two distinct elements $x_1, x_2 \in A$,

[1] Fixing n as a system-wide parameter also avoids problems arising from using parameters of different length.

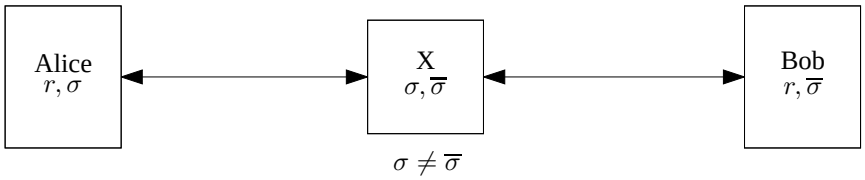


Fig. 1. An adversary X sitting between Alice and Bob. r is a pre-shared secret, σ and $\overline{\sigma}$ are secrets established by means of quantum cryptography.

there are at most $\varepsilon |\mathcal{H}|$ functions $h \in \mathcal{H}$ such that $h(x_1) = h(x_2)$. Such a class is abbreviated as ε-AU_2

Definition 3.2: Let $\varepsilon > 0$ be a real number. $\mathcal{H}$ is called *ε-almost strongly universal$_2$* (or ε-ASU_2), if

1) for any $x \in A, y \in B$ there are exactly $\frac{|\mathcal{H}|}{|B|}$ functions $h \in \mathcal{H}$ such that $h(x) = y$.
2) for any two distinct elements $x_1, x_2 \in A$, and for any two elements $y_1, y_2 \in B$, there are at most $\varepsilon \frac{|\mathcal{H}|}{|B|}$ functions $h \in \mathcal{H}$ such that $h(x_1) = y_1$ and $h(x_2) = y_2$.

The conditions for ε-almost strongly universal$_2$ directly map to the probabilities for impersonation and substitution, thus if we have an ε-ASU_2 class of hash-functions, we can use them for unconditionally secure authentication.

Having an ε-ASU_2 hash-family, we can easily construct an authentication code considering the following result:

Theorem 3.1 (Theorem 3.2 in [18]): If there exists an ε-ASU_2 class $\mathcal{H}$ of hash-functions from A to B, then there exists an authentication code for $|A|$ source states having $|B|$ authenticators and $|\mathcal{H}|$ authentication rules, such that $P_I = \frac{1}{|B|}$ and $P_S \leq \varepsilon$.

We will use the following hash-function: Agree on a prime power q and two secrets $a, b \in GF(q)$ prior to any authentication. The authentication code for any message $m \in GF(q)$ is calculated as

$$h(m) = a \cdot m + b, \tag{1}$$

where $h(m) \in GF(q)$. Henceforth, this message m will be the secret Alice and Bob agreed on, using quantum cryptography, thus we set $m := \sigma$ or $\overline{\sigma}$, respectively. Equation (1) is a linear function and is uniquely determined if and only if two points are given. An eavesdropping adversary will only learn $(\sigma, h(\sigma))$ which can be considered as one point on the function. Obviously, it is impossible to reconstruct the function using this knowledge, so extraction of the secret (a, b) and therefore forging subsequent messages is impossible as long as the secret (a, b) is *discarded* after being used *once*. Otherwise, an adversary obtains two values $z_1 = a \cdot \sigma_1 + b$ and $z_2 = b \cdot \sigma_2 + b$ from which a and b are obtained immediately. Multiple authentication depends on whether the last authentication was successful or not:

- If the authentication was successful, then even an infinitely powerful passive adversary cannot learn anything about the secret a, b and, by the properties of quantum cryptography (BB84 for instance), Alice and Bob possess perfectly secure identical secrets, from which they can derive new secrets a' and b'. So by the time a passive adversary could try to extract information, it has become outdated and worthless.
- Authentication may fail either if Eve was sitting in the middle or if some distortion from outside or malfunction of some intermediate device caused modification or loss of z. In this case, Alice and Bob switch to new secrets such that an eavesdropping adversary has no chance of revealing a or b. This is addressed in the next section.

B. Multiple Authentication

Suppose an adversary has been detected and the existing secrets a, b have already been used once. The problem is the determination of new secrets such that subsequent authentications are possible. Ideally, Alice and Bob should establish new secrets without any interaction, so Eve has no hope to penetrate the message-flow for influencing the secrets. Suppose we would apply a function $u = (u_1, u_2)$ to the pair (a, b) for creating a new secret (a', b'). Then Eve gets the values

$$\begin{aligned} z &= a \cdot \sigma + b \\ z' &= a' \cdot \sigma' + b' = u_1(a, b) \cdot \sigma' + u_2(a, b), \end{aligned}$$

which is a system of two equations for two unknowns. Here, the key σ' has been established by Alice and Bob for another authentication round. Even if there does not exist an analytic solution to this system, Eve may be still be able to obtain a partial or approximate solution. Intuitively, it is therefore not possible to create (a', b') from (a, b) such that the new (a', b') is stochastically independent of (a, b), because if this was possible, we could generate a truly random sequence only by computational means.

We can permit re-using the secrets *a certain number of times* before Alice and Bob need to re-initialize their authentication engines by getting in touch with each other personally. Since we cannot hope to create truly new secrets from old ones, we need more pre-shared secrets. Let $C = \{d_1, d_2, \ldots, d_l\}$ denote l pre-shared $2n$-bit secrets, which are uniformly distributed, stochastically independent and ordered. Let 2^C denote the power-set of C and let $D_1, D_2, \ldots, D_{2^l-1}$ be an arbitrary enumeration of the $t = 2^l - 1$ *non-empty* subsets in 2^C, which is known to Alice and Bob. For the i-th authentication trial, select a set $D_i \in 2^C$ and define

$$r_i := a_i || b_i = \bigoplus_{d \in D_i} d, \tag{2}$$

i.e. apply the $\oplus$-operation to all elements in D_i and partition the result to find a_i and b_i, both of which have length $|a_i| =$

$|b_i| = n$ for all $i = 1, \ldots, 2^l - 1$. The secret $r_i := (a_i, b_i)$ can be used for creating the Wegman-Carter MAC. More compactly, we create a set

$$R = \left\{ (a,b) \,\middle|\, \exists D \subseteq C, D \neq \emptyset : a||b = \bigoplus_{d \in D} d \right\} = \{r_1, \ldots, r_t\}$$

of secrets. For unconditional security, we require stochastic independence of different secrets used for the different authentication trials.

We first give the protocol and let the formal justification follow:

Protocol

Initialization: Pre-distribute $l \geq 2$ secret strings $d_1, d_2, \ldots, d_l \in \{0,1\}^{2n}$ (by non-cryptographic means (Smartcards, etc.) for instance) and create secrets (a_i, b_i) according to equation (2). Let $h_i(m) := a_i m + b_i$ be the hash-function using secret (a_i, b_i).

Protocol actions:

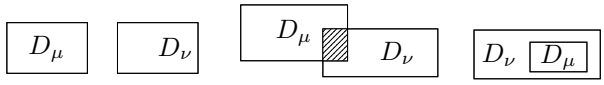
1) Using quantum cryptography, Alice exchanges a key σ with another person P which she thinks is Bob. Bob establishes a key $\overline{\sigma}$ with a person Q which he thinks is Alice. The case $P = Q =$ Eve is possible, as well as the situation $P =$ Bob and $Q =$ Alice (i.e. no adversary). Both partition the prefix of their key σ (or $\overline{\sigma}$, respectively) as $\sigma = i||j||k_1||k_2||\cdots$, with $|i| = |j| = l, |k_1| = |k_2| = n$.
2) Alice selects the secret (a_i, b_i) for the MAC and sends $s_A = h_i(k_1)$ to Bob.
3) Bob does as Alice by doing the same partitioning on his key $\overline{\sigma}$ (yielding values $\overline{i}, \overline{j}, \overline{k}_1$, and $\overline{k}_2$) and verifies $s_A \stackrel{?}{=} h_{\overline{i}}(\overline{k}_1)$. Upon equality, he responds with $\overline{s}_B = h_{\overline{j}}(\overline{k}_2)$. If verification fails, he stops executing the protocol.
4) Alice verifies $h_j(k_2) \stackrel{?}{=} \overline{s}_B$ for her part k_2, where $\overline{s}_B$ is the message she received from "Bob". She accepts if equality holds.
5) If both parties accept, they extract portions $(d'_1, d'_2, \ldots, d'_l)$ from their common key $\sigma = \overline{\sigma}$ and use them for subsequent authentications.
6) If one party rejects the authentication, then both may re-try up to $t = 2^l - 2$ times before there is a need to re-initialize the protocol.

Lemma 3.2: Let $n > 0$, let $x_1, \ldots, x_l \in \{0,1\}^n$ be independent and uniformly distributed random strings, and let J be a non-empty subset of $I = \{1, \ldots, l\}$. Create the set $R := \left\{ r_J = \bigoplus_{j \in J} x_j | J \subseteq I, J \neq \emptyset \right\}$. Then the strings in every proper subset of R are stochastically independent.

Proof: Define $C := \{x_1, \ldots, x_l\}$ and let $A \triangle B := (A \backslash B) \cup (B \backslash A)$ denote the *symmetric difference* of the sets A and B, i.e. we have $A \triangle B = \emptyset \iff A = B$. Let r_μ have been created from a non-empty subset $D_\mu \subseteq C$ and let r_ν have been created from a different and non-empty subset $D_\nu \subseteq C$, according to equation (2). From $D_\nu \neq D_\mu$ it follows that $M := D_\mu \triangle D_\nu \neq \emptyset$, so without loss of generality, r_μ can be viewed as a one-time pad encrypted version of r_ν, i.e.

$$r_\mu = r_\nu \oplus \overbrace{\left(\bigoplus_{x \in M} x \right)}^{\text{O.T.P.}}. \tag{3}$$

Hence, r_μ is stochastically independent of r_ν, as $\bigoplus_{x \in M} x$ is independent of r_μ, r_ν. (Equation (3) can be seen immediately considering the three possible situations depicted in figure 2.)



(a) Disjoint sources (b) Intersecting sources (c) Overlapping sources

Fig. 2. Sources for Wegman-Carter secrets. r_μ obtained from r_ν by one-time pad encrypting with the symmetric difference $D_\mu \triangle D_\nu$.

Consider the joint distribution $\Pr(r_1, \ldots, r_t)$ and re-write it using conditional probabilities as

$$\Pr(r_1, \ldots, r_t) = \prod_{i=0}^{t-1} \Pr(r_{t-i} | r_1, \ldots, r_{t-i-1}). \tag{4}$$

Look at a function $\varphi : 2^C \to 2^C$ and assume that the power-set of C can be enumerated by repeated application of φ. In other words, the finite sequence $\left\{\varphi^i(\emptyset)\right\}_{i=1}^{t}$ shall coincide with $2^C \backslash \{\emptyset\} = \{D_1, D_2, \ldots, D_t\}$. We then have D_i independent of D_j for any i, j with $j < i - 1$, as D_i is a function of D_{i-1} and therefore stochastically independent of any other subset different from D_{i-1}. Since r_i is created from D_i, we have $\Pr(r_i | r_1, \ldots, r_{i-1}, r_{i+1}, \ldots, r_t) = \Pr(r_i | r_{i-1})$. The r_i's thus obey a first-order Markov-property, so equation (4) reduces to

$$\Pr(r_1, \ldots, r_t) = \Pr(r_1) \prod_{i=1}^{t-1} \Pr(r_{i+1} | r_i),$$

as the next number is only dependent on its predecessor. But the numbers are pairwise independent (with $\mu = i+1, \nu = i$), as we have shown above, so we finally get the factorization

$$\Pr(r_1, \ldots, r_t) = \Pr(r_1) \prod_{i=1}^{t-1} \underbrace{\Pr(r_{i+1} | r_i)}_{= \Pr(r_{i+1})} = \prod_{i=1}^{t} \Pr(r_i),$$

hence the numbers in R are stochastically independent. It remains to specify a function φ such that D_{i+1} can be generated from D_i (and solely from D_i), and such that $2^C \backslash \{\emptyset\}$ is fully enumerable by repeated application of φ. Let $\psi : 2^C \to \mathbb{N}$ be defined as

$$\psi(D) = \sum_{i=1}^{l} 2^{i-1} \left| D \cap \{x_i\} \right|,$$

then it is easy to see that $\psi(D)$ returns value whose binary representation is a string being $l = |C|$ bits long, and having a "1" at position i iff $d_i \in D$, and "0" otherwise. The set D_{i+1} is created from D_i by applying the function φ, defined as

$$\varphi(D) := \psi^{-1}(1 + \psi(D)),$$

for any subset $D \subseteq C$. The function φ is independent of i (stationary iteration scheme), and has the desired properties, as $\varphi(\emptyset) = x_1$ and we explore all values from $1, \ldots, 2^l$, thus get different output sets, none of which is empty. It is obvious that we eventually enumerate all subsets of C by this iteration scheme, so $2^C \setminus \{\emptyset\} = \bigcup_{i=1}^{2^l} \{\varphi^i(\emptyset)\}$. As $\left|2^C \setminus \{\emptyset\}\right| = 2^l - 1$, we can create $2^l - 1$ strings r_i before re-using a secret. ■

Let X, Y be uniformly and independently distributed random variables with domain $GF(p^n)$ for p being a prime and n being an integer. Moreover, let $q = |GF(p^n)|$.

We have $\Pr(X + Y = z)$ for $z \in GF(p^n)$ as $\Pr(X + Y = z) = \sum_{j \in GF(p^n)} \Pr(X = j, Y = z - j) = \sum_{j \in GF(p^n)} \Pr(X = j) \Pr(Y = z - j) = qp^2$ with $p = \Pr(X = j) = \Pr(Y = z - j) = 1/q$ following from the independence of X, Y and uniformity of X, Y on $GF(p^n)$. We finally get $\Pr(X + Y = z) = 1/q$ for any $z \in GF(p^n)$. A similar argument can be given for $\Pr(X \cdot Y = z)$ as also $X = j, Y = z/j$ will lie in $GF(p^n)$, thus possess a nonzero probability, so we also get $\Pr(X \cdot Y = z) = 1/q$. We conclude that for three independent and uniformly distributed random variables X, Y, Z we have $W = X \cdot Y + Z$ also uniformly distributed on $GF(p^n)$.

Theorem 3.3: If Alice and Bob share a set $C = \{d_1, d_2, \ldots, d_l\}$ of l independent, uniformly distributed, and secret random strings $d_i \in \{0,1\}^{2n}$, then an active meet-in-the-middle adversary has a success probability of at most 2^{-n}, if the protocol fails no more than $2^l - 2$ times.

Proof: Consider two MACs $r_1 k + r_2$ and $r_1' k' + r_2'$ for two random strings k, k' and four numbers r_1, r_2, r_1', r_2' created using lemma 3.2. We have $\Pr(r_1 k + r_2 | r_1' k' + r_2') = \frac{\Pr(r_1 k + r_2, r_1' k' + r_2')}{\Pr(r_1' k' + r_2')}$, where $r_1 k + r_2$ is independent of $r_1' k' + r_2'$ considering the independence of r_1, r_2, r_1', r_2' (lemma 3.2) and the independence of the QKD-keys k, k' together with the preceding discussion. Thus $\Pr(r_1 k + r_2, r_1' k' + r_2) = \Pr(r_1 k + r_2) \Pr(r_1' k' + r_2')$ and the conditional probability is $\Pr(r_1 k + r_2 | r_1' k' + r_2') = \Pr(r_1 k + r_2)$. So different authentication trials do not provide any information about each other and the adversaries success probability follows from the properties of the Wegman-Carter MAC. The bound on the number of retrials is obtained by recalling that no information is leaking unless *all* $2^l - 1$ secrets have been used. ■

After the $(2^k - 2)$-th failure of the protocol, re-initialization by non-cryptographic means is necessary.

Example 3.1: Choosing $l = 20$ and $n = 128$ allows for $2^{20} - 2 = 1048574$ trials after an authentication failed and before manual re-initialization is necessary. We have a 128-bit MAC and $l \cdot 2n = 5120$ bits of authentication data. This data needs to be re-negotiated after any authentication, hence we need $2 \cdot 128$ bits for bidirectional authentication plus 5120 bit for the new secrets, which implies a minimum key-length of 5376 bits (= 672 bytes), which is also the amount of data transmitted for authentication with key-updating afterwards. The success probability for an adversary under this setup is at most 2^{-128}.

C. Authenticating Payload

If the authentication succeeds, and we have re-negotiated the keys C' for subsequent trials, then we may use the existing (old) material C for authenticating messages. As before, we can use the secret-set R (created from C) for authenticating messages of length n bit using a Wegman-Carter MAC. To do so, we first partition the message into blocks m_i and hash each block so it is n bits long. A MAC for the result is obtained in the usual way using our secrets from lemma 3.2. Unconditional security can be established upon considering the following theorem:

Theorem 3.4 ([18], Theorem 5.5): Suppose $\mathcal{H}_1$ is an ε_1-AU_2 class of hash-functions from A to B and $\mathcal{H}_2$ is an ε_2-ASU_2 class of hash-functions from B to C. Then there exists an ε-ASU_2 class $\mathcal{H}$ of hash-functions from A to C, where $\varepsilon = \varepsilon_1 + \varepsilon_2$ and $|\mathcal{H}| = |\mathcal{H}_1| \cdot |\mathcal{H}_2|$.

Applied to our problem, this means that we first may hash (compress) each block m_i using any almost two-universal hash function $\widetilde{h}$ with output length n and create a Wegman-Carter MAC for m_i as $h(m_i) = a_i \widetilde{h}(m_i) + b_i$. It then follows from theorem 3.4 that the scheme remains secure. Theorem 3.1 then provides bounds on the probabilities for attacks to be successful. [18] provides an efficient construction suitable for our needs.

D. The Requirement $\sigma \neq \overline{\sigma}$

As previously pointed out, our authentication scheme relies on the assumption that Alice and Bob have exchanged *distinct* keys with Eve by means of quantum cryptography.

Take a look at the BB84 protocol [2], [16]. The details of each step shall not be of interest for us, so we restrict ourselves to a protocol sketch here:

1) Exchange polarized photons.
2) Publicly communicate the used polarization planes.
3) Do error correction and measurement as well as privacy amplification.

It is especially the *privacy amplification* we are interested in: Roughly speaking, privacy amplification is done by applying a hash-function to the raw key, which is partially secret to have a shorter key which is "sufficiently" secret. Sophisticated theoretical results are provided by [14], [9] to mention just two. The problem with this generic approach is that if the key is fully under Eve's control until the privacy amplification, she can choose the bits of the raw key such that the hashing produces something she likes. One mechanism for privacy amplification is combining pairs of bits using the XOR such that Shannon entropy of Eve's information is sufficiently increased, or equivalently, her information about the key is lowered below an acceptable threshold. If we have Alice and Bob (one of them could possibly be Eve) agree on the bits by alternatingly issuing indices of which bits to use, then combining these bits using the XOR operation, both get a key over which none of them has full control. The point here is that neither Alice nor Bob fully specify which bits to combine. Rather, Alice announces i_1 and Bob announces i_2, so the final

bit at the, say j-th, position is $b_{i_1} \oplus b_{i_2}$. Although none of the parties has full control over the generated key, but the keys may nevertheless coincide. Unfortunately, without interaction we cannot guarantee distinctness, so the best we can do, is giving the probability for such an accidental coincidence. As σ and $\overline{K}$ can be viewed as independent and uniformly distributed random numbers, we have a probability of coincidence being $\Pr(\sigma = \overline{\sigma}) = \Pr(\sigma = k)\Pr(\overline{\sigma} = k) = (2^{-2n})^2 = 2^{-4n}$. For our example, this means that the requirement $\sigma \neq \overline{\sigma}$ is violated with a probability of $2^{-4 \cdot 128}$ which is clearly negligible.

E. Synchronization

If an adversary is somehow able to influence the system of one participant, Alice and Bobs systems may become unsynchronized, meaning that different authentication secrets are selected and subsequent authentication trials fail. The partitioning $\sigma = i||j|| \cdots$ (cf. protocol step 1) explicitly yields random and identical indices i and j on each side determining the secrets to be used, so no communication or local maintenance of indices is required. Hence, an adversary cannot fiddle with synchronization of Alice and Bob. However, Alice and Bob still need to do some bookkeeping to avoid re-using secrets. This can be achieved robustly by deleting secrets from R after each authentication trial.

IV. Replacing QKD

Working through the formal arguments in the previous paragraphs, we see that we actually did not rely on any particular property of quantum cryptography except for its information-theoretic security. Relaxing this requirement allows for replacement of the quantum cryptographic key-exchange by an arbitrary key-establishment protocol, as long as the resulting key is hashed according to the remarks in section III-D afterwards (to prevent identical keys if Eve is present between Alice and Bob). One candidates for a substitution of QKD is the Diffie-Hellman Protocol [3].

V. Conclusion

The authentication scheme given in this report relies on the assumption that key-agreement cannot be controlled by any of the participating parties. If this holds, then we are able to authenticate Alice and Bob while being able to detect an active adversary. With l pre-shared secrets, our scheme is resistant against meet-in-the-middle attacks with a probability $\leq 2^{-n}$ for n-bit authentication secrets. Moreover, the scheme remains secure up to $2^l - 2$ times even if an attacker is detected. The protocol is generic, thus can be instantiated with any form of key-agreement, not necessarily BB84 or Diffie-Hellman.

If an authentication fails, other messaging paths distinct from the one that failed, may be tried. This requires the network to be "sufficiently connected", i.e. to provide a sufficiently large number of non-intersecting communication paths between any two nodes [13]. Moreover, the protocol can be used for authentication in the context of unconditionally secure message-transfer. This was an open issue in [13].

References

[1] C. Bennet. Quantum cryptography: Uncertainty in the service of privacy. *Science*, 257(7):752–753, 1992.

[2] C. Bennet and G. Brassard. Quantum cryptography: Public key distribution and coin tossing. In *Proc. IEEE Int. Conference on Computers, Systems, and Signal Processing*, page 175, Bangalore, 1984.

[3] W. Diffie and M. Hellman. New directions in cryptography. *IEEE Transactions on Information Theory*, IT-22(6):644–654, 1976.

[4] U. Feige, A. Fiat, and A. Shamir. Zero-knowledge proofs of identity. *J. Cryptology*, 1(2):77–94, 1988.

[5] S. Ghernaouti-Hélie and M. Sfaxi. Upgrading PPP security by quantum key distribution. In *NetCon 2005 conference*, 2005.

[6] S. Ghernaouti-Hélie, M. Sfaxi, G. Ribordy, and O. Gay. Using quantum key distribution within IPSEC to secure MAN communications. In *MAN 2005 conference*, 2005.

[7] L. C. Guillou and J.-J. Quisquater. A practical zero-knowledge protocol fitted to security microprocessors minimizing both transmission and memory. In C. G. Gunther, editor, *In Advances in Cryptology — EUROCRYPT '88*, volume 330 of *LNCS*, pages 123–128. Springer–Verlag, 1988.

[8] T. Helleseth and T. Johansson. Universal hash functions from exponential sums over finite fields and galois rings. In *CRYPTO '96: Proceedings of the 16th Annual International Cryptology Conference on Advances in Cryptology*, pages 31–44, London, UK, 1996. Springer-Verlag.

[9] R. Koenig, U. Maurer, and R. Renner. On the power of quantum memory. *IEEE Transaction on Information Theory*, 51(7):2391–2401, July 2005.

[10] H. Krawczyk. LFSR-based hashing and authentication. In *CRYPTO '94: Proceedings of the 14th Annual International Cryptology Conference on Advances in Cryptology*, pages 129–139, London, UK, 1994. Springer-Verlag.

[11] K. Mehlhorn and U. Vishkin. Randomized and deterministic simulations of PRAMs by parallel machines with restricted granularity of parallel memories. *Acta Inf.*, 21(4):339–374, 1984.

[12] A. Menezes, P. C. van Oorschot, and S. Vanstone. *Handbook of applied Cryptography*. CRC Press LLC, 1997.

[13] S. Rass. How to send messages over quantum networks in an unconditionally secure manner. Technical Report TR-syssec-05-05, University of Klagenfurt, Computer Science, System Security, Klagenfurt, September 2005.

[14] R. Renner and R. Koenig. Universally composable privacy amplification against quantum adversaries. In J. Kilian, editor, *2nd Theory of Cryptography Conference, TCC 2005*, volume 3378 of *LNCS*, pages 407–425. Springer, Feb. 2005.

[15] P. Rogaway. Bucket hashing and its application to fast message authentication. *Journal of Cryptology*, 12(2):91–115, 1999.

[16] P. Shor and J. Preskill. Simple proof of security of the BB84 quantum key distribution protocol. *Phys. Rev. Lett.*, 85:441–444, 2000.

[17] V. Shoup. On fast and provably secure message authentication based on universal hashing. In *CRYPTO '96: Proceedings of the 16th Annual International Cryptology Conference on Advances in Cryptology*, pages 313–328, London, UK, 1996. Springer-Verlag.

[18] D. R. Stinson. Universal hashing and authentication codes. In *CRYPTO '91: Proceedings of the 11th Annual International Cryptology Conference on Advances in Cryptology*, pages 74–85, London, UK, 1992. Springer-Verlag.

[19] M. Wegman and J. Carter. Universal classes of hashing functions. *Journal of Computer and System Sciences*, 22:265–279, 1981.

[20] M. Wegman and L. Carter. New hash functions and their use in authentication and set equality. *Journal of Computer and System Sciences*, 22:265–279, 1981.