An Optimal Non-interactive Message Authentication Protocol

Sylvain Pasini and Serge Vaudenay

EPFL, CH-1015 Lausanne, Switzerland
http://lasecwww.epfl.ch

Abstract. Vaudenay recently proposed a message authentication protocol which is interactive and based on short authenticated strings (SAS). We study here SAS-based non-interactive message authentication protocols (NIMAP). We start by the analysis of two popular non-interactive message authentication protocols. The first one is based on a collision-resistant hash function and was presented by Balfanz et al. The second protocol is based on a universal hash function family and was proposed by Gehrmann, Mitchell, and Nyberg. It uses much less authenticated bits but requires a stronger authenticated channel.

We propose a protocol which can achieve the same security as the first protocol but using less authenticated bits, without any stronger communication model, and without requiring a hash function to be collision-resistant. Finally, we demonstrate the optimality of our protocol.

1 Introduction

Message authentication protocols are typically used to exchange public keys so that *secure* communications can be set up. For a better usability, a non-interactive protocol is preferred. It should be noted that the protocol uses two separate channels. The first one is a broadband insecure channel (e.g. an email or a wireless channel) and the second one is a narrowband authenticated channel (e.g. authentication by a human voice or a manual authentication by a human operator).

In SSH and in GPG, the simple folklore protocol used to exchange public keys is presented in Balfanz et al. [BSSW02]. It is non-interactive and based on a collision-resistant hash function. The authenticated string is the k-bit hashed value of the input message m. We recall that this protocol is typically weak against offline attacks, such as birthday attacks, which have a complexity of $2^{k/2}$ and that hash functions which resist to collision attacks are threatened species these days [BCJ+05, WLF+05, WYY05b, WYY05a, WY05]. For instance, it is possible to forge two different RSA keys with the same MD5 hash as shown in [LWdW05, LdW05].

Another protocol is MANA I which was proposed by Gehrmann-Mitchell-Nyberg [GMN04]. It is based on an universal hash function family. This protocol is more resistant against offline attacks since it uses an authenticated value which has a random part K. The second part is the hashed value (using K as key) of the input message m. The protocol requires to send the hashed value "at once". Hence, even if an adversary has an infinite complexity, his probability of success is at most 2^{-k} where k is the size

D. Pointcheval (Ed.): CT-RSA 2006, LNCS 3860, pp. 280–294, 2006.

of K and the size of the hash. However, the requirement renders the protocol "less non-interactive" by imposing a strong assumption on the communication model.

We propose a protocol which has the same security than the one presented by Balfanz et al. [BSSW02] but using less authenticated bits and without requiring the hash function to be collision-resistant. Our protocol is based on a trapdoor commitment scheme in the Common Reference String (CRS) model or in the Random Oracle model.

Finally, we propose a definition of the optimality of a message authentication protocol and we analyze the three above protocols.

2 Preliminaries

The considered model is a communication network made up of devices which use insecure broadband communication channels between them. In addition, they can use a narrowband channel which can be used to authenticate short messages, i.e. short authenticated strings (SAS).

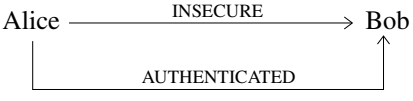


Fig. 1. NIMAP Channels

Communication devices are located on nodes n of given identity ID_n and can run several instances which are formally denoted by a unique instance tag π_n^i. We concentrate on non-interactive message authentication protocols (NIMAP).

2.1 Adversarial Model Against NIMAP

A *message authentication protocol* has an input m on the side of the claimant Alice of identity ID and an output $\widehat{\mathsf{ID}}||\hat{m}$ on the side of the verifier Bob. Authentication is successful if the output is $\widehat{\mathsf{ID}} = \mathsf{ID}$ and $\hat{m} = m$. The protocol is non-interactive if it only uses messages send by Alice to Bob.

We assume that adversaries have full control on the broadband communication channel. Indeed, an attacker can read messages from the channel, he can prevent a message from being delivered, he can delay it, replay it, modify it, and change its recipient address. Here, we adopt the security model from Vaudenay [Vau05] based on Bellare-Rogaway [BR93]. The adversary has full control on which node launches a new instance of a protocol, on the input of the protocol, and on which protocol instance runs a new step of the protocol. Namely, we assume that the adversary has access to a $\mathsf{launch}(n,r,x)$ oracle where n is a node, r is a character, Alice or Bob, and x is the input. This oracle returns a unique instance tag π_n^i. Since a node can a priori run concurrent protocols, there may be several instances related to the same node n. The adversary also has access to the oracle $\mathsf{receive}(\pi_A^i)$ which returns a message m which is meant to be sent to Bob and to the oracle $\mathsf{send}(\pi_B^i, m)$ which sends a message m to a given instance of Bob.

Typically, a NIMAP between nodes A and B with input m on the side of Alice and using two messages runs as follows.

1. $\pi_A \leftarrow \mathsf{launch}(A, \mathrm{Alice}, m)$
2. $p_1 \leftarrow \mathsf{receive}(\pi_A)$
3. $p_2 \leftarrow \mathsf{receive}(\pi_A)$
4. $\pi_B \leftarrow \mathsf{launch}(B, \mathrm{Bob}, \emptyset)$
5. $\mathsf{send}(\pi_B, p_1)$
6. $\widehat{\mathsf{ID}}||\hat{m} \leftarrow \mathsf{send}(\pi_B, p_2)$

By convention, we describe protocols by putting a *hat* on the notation for Bob's received messages (i.e. inputs of the send oracle) which are not authenticated since they can differ from Alice's sent messages (i.e. outputs of the receive oracle) in the case of an active attack.

On a global perspective, several $\mathsf{launch}(A_k, \mathrm{Alice}, m_k)$ and $\mathsf{launch}(B_\ell, \mathrm{Bob}, \emptyset)$ can be queried. These queries create several $\pi_{A_k}^{i_k}$ instances of Alice (authentication claims) and several $\pi_{B_\ell}^{j_\ell}$ instances of Bob (authentication verifications). We may have a perfect matching between the k's and ℓ's such that related instances have matching conversations which fully follow the protocol specifications, and the $\pi_{B_\ell}^{j_\ell}$ ends with output $\mathsf{ID}_{A_k}||m_k$ for the matching k. In any other case, we say that an attack occurred. We say that an *attack is successful* if there exists at least an instance $\pi_{B_\ell}^{j_\ell}$ which terminated and output $\widehat{\mathsf{ID}}||\hat{m}$ such that there is no k for which $\widehat{\mathsf{ID}} = \mathsf{ID}_{A_k}$ and $\hat{m} = m_k$. Note that many protocol instances can endlessly stay in an unterminated state or turn in an abort state. We call *one-shot attacks* the attacks which launch a single instance of Alice and Bob. The *attack cost* is measured by

- the number Q of launched instances of Alice, i.e. the *online complexity*.
- the additional complexity C, i.e. the *offline complexity*.
- the probability of success p.

Here is a useful lemma taken from [Vau05].

Lemma 1. *We consider a message authentication protocol with claimant Alice and verifier Bob in which a single SAS is sent. We denote by μ_A (resp. μ_B) the complexity of Alice's (resp. Bob's) part. We consider adversaries such that the number of instances of Alice (resp. Bob) is at most Q_A (resp. Q_B). We further denote T_0 and p_0 their time complexity and probability of success, respectively. There is generic transformation which, for any Q_A, Q_B, and any adversary, transforms it into a one-shot adversary with complexity $T \leq T_0 + \mu_A Q_A + \mu_B Q_B$ and probability of success $p \geq p_0/Q_A Q_B$.*

Assuming that no adversary running a one-shot attack has a probability of success larger than p, using Lemma 1, we can upper bound the probability of success of an attack which uses Q_A, resp. Q_B, instances of Alice, resp. Bob, by $Q_A Q_B p$.

2.2 Authenticated Channels

When referring to "channel", we refer by default to an insecure broadband channel without any assumption. As mentioned before, the devices can use an authenticated channel. An *authenticated channel* is related to a node identity ID. Formally, an authenticated channel from a node n has an identifier ID_n. It allows the recipient of a message to know the identity of the node from which the message has been sent as is. Note that

an adversary cannot modify it (i.e. integrity is implicitly protected), but she can delay it, remove it, or replay it, and of course, read it. Precisely, an authenticated channel does not provide confidentiality. By convention, we denote $\mathsf{authenticate}_{\mathsf{ID}_n}(x)$ a message x which has been sent from node n through the authenticated channel.

The receive oracle maintains unordered sets of authenticated messages in every channel ID_n from node n. Only receive oracles with a π_n^i instance can insert a new message in this set. When a send oracle is queried with any message $\mathsf{authenticate}_{\mathsf{ID}_n}(x)$, it is accepted by the oracle only if x is in the set related to channel ID_n. Note that concurrent or successive instances related to the same node write in the same channel, i.e. in the same set. Thus, when an instance of Alice sends a message, Bob can only authenticate the node from which it has been sent, i.e. n, but not the connection to the right instance.

Weak Authenticated Channels. By default, authenticated channels without any other assumption are called *weak*. This means that an adversary can delay a message, remove it, or replay it. In particular, the owner of the message has not the insurance that the message has been delivered to the recipient.

Stronger Authenticated Channels. In some cases we need special assumptions on the authenticated channel. We can consider *stronger authentication channels*, namely channels in which additional properties are achieved as proposed by Vaudenay [Vau05]. In the following, we use one possible property that can be assumed on a stronger authentication channel. A *stall-free transmission* assumes that when a message is released by a receive oracle either it is used as input in the immediately following send oracle query or it is never used. Namely, we cannot wait for a new message from Alice before delivering the authenticated message to Bob.

For instance, a *face to face conversation* and a *telephone call* are clearly authenticated channels. When one talks to the other one, the recipient further knows that the message has not been recorded since interactivity implies coherent conversations (stall-free). *Mail*, *e-mail*, and *voice mail* can be stalled and released in a different order. Note that an e-mail without any cryptographic appendix such as a GPG signature is in fact not an authenticated channel since it can easily be forged.

2.3 Hash Functions

Collision-Resistant Hash Functions (CRHF). A collision-resistant hash function is a hash function in which it should be hard to find two inputs x and y such that $H(x) = H(y)$ and $x \neq y$. Due to the birthday attacks, the hash length must be at least of 160 bits.

Weakly Collision-Resistant Hash Functions (WCRHF). Weak collision resistance means that the game of Fig. 2 is hard. Assume a (T, ε)-weakly collision-resistant hash function H defined on a finite set $\mathcal{X}$. Any adversary $\mathcal{A}$ bounded by a complexity T wins the WCR game on Fig. 2 with probability at most ε.

Universal Hash Functions Families (UHFF). An ε-universal hash function family is a collection of functions H_K from a message space to a finite set $\{0,1\}^k$ which depends on a random parameter K such that for any $x \neq y$ we have

$$\Pr[H_K(x) = H_K(y)] \leq \varepsilon$$

where the probability is over the random selection of K.

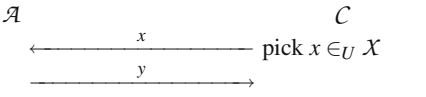


Fig. 2. WCR game

2.4 Commitment Schemes

We can formalize a *commitment scheme* by two algorithms commit and open. For any message m we have $(c,d) \leftarrow \mathsf{commit}(m)$. The c value is called the *commit* value and the d value the *decommit* value. Knowing both c and d, the message can be recovered using the open oracle, i.e. $m \leftarrow \mathsf{open}(c,d)$. Intuitively, a commitment scheme should be *hiding*, meaning that for any c, it is hard to deduce any information about the corresponding message m, and *binding*, meaning that one cannot find c,d,d' such that (c,d) and (c,d') open to two different messages. We also introduce *keyed commitment schemes* which have in addition a setup oracle to initialize a pair of keys, i.e. $(K_p,K_s) \leftarrow \mathsf{setup}()$. The public key K_p is used in commit and open oracles. Keyed commitment schemes should be understood as working in the Common Reference String (CRS) model. Namely, K_p is a common reference string set up once for all and K_s is unknown to anyone.

Binding Property. The semantic binding (SB) game of Fig. 3 must be hard, i.e. for any message m and any commit value c one cannot find two decommit values d and d' such that $m \leftarrow \mathsf{open}(K_p,c,d)$ and $m' \leftarrow \mathsf{open}(K_p,c,d')$ with $m \neq m'$. The scheme is (T,ε)-semantically binding if any adversaries $\mathcal{A}$ bounded by a complexity T has a probability to find two decommit values d and d' which is at most ε.

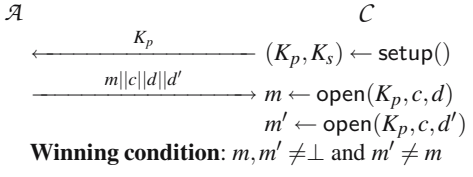


Fig. 3. SB Game

Trapdoor Commitment Model. The notion of *trapdoor commitment* was introduced by Brassard, Chaum, and Crepeau [BCC88]. We define (T,ε)-trapdoor commitment schemes by four algorithms setup, commit, open, and equivocate. The first three work as before. The algorithm equivocate defeats the binding property by using the secret key K_s. More precisely, for any $(K_p,K_s) \leftarrow \mathsf{setup}()$ we have

- for any m and any $(c,d) \leftarrow \mathsf{commit}(K_p,m)$ we have $m \leftarrow \mathsf{open}(K_p,c,d)$,
- for any m, by running $(c,d) \leftarrow \mathsf{commit}(K_p,m)$, c is uniformly distributed,
- for any m, any $\hat{c}$, and any $\hat{d} \leftarrow \mathsf{equivocate}(K_s,m,\hat{c})$, the $\mathsf{open}(K_p,\hat{c},\hat{d})$ algorithm yields m.
- for any adversary bounded by a complexity T in the SB game, the winning probability is smaller than ε.

Note that this primitive is a particular case of *strongly equivocable commitment* as described by Damgård-Groth [DG03].

Trapdoor commitment schemes are perfectly hiding and computationnaly binding commitment schemes. Note that for any (K_p, K_s) and any m, the distribution of (c,d), which has been yield using the commit algorithm, is equal to the distribution of $(\hat{c},\hat{d})$, which have been yield choosing a $\hat{c}$ with uniform distribution and using the equivocate algorithm.

For instance, a trapdoor commitment based on the discrete logarithm problem was proposed by Boyar and Kurtz [BK90]. Another trapdoor commitment scheme was proposed by Catalano et al. [CGHGN01] based on the Paillier's trapdoor permutation [Pai99]. The proposed scheme uses an RSA modulus $N = pq$ and a value $h \in \mathbb{Z}_{N^2}$ such that its order is a multiple of N. The public key is $K_p = (N,h)$ and the private key is $K_s = (p,q)$. The commit algorithm of a message m picks uniformly two random values r,s and outputs $c \leftarrow (1+mN)r^N h^s \bmod N^2$ and $d = (r,s)$. Note that the commit value c is uniformly distributed for any m since r and s are uniformly distributed and $(r,s) \mapsto r^N h^s \bmod N^2$ is the Paillier trapdoor permutation (see [Pai99]). We denote $\mathcal{F}_h(r,s)$ this permutation. The decommit algorithm simply checks that $c = \mathsf{commit}(K_p, m)$ with $d = (r,s)$. The trapdoor is the collision-finding function: given a commit $\hat{c}$ and a message m, one can find $\hat{d} = (\hat{r},\hat{s})$ such that $\hat{c} = (1+mN)\mathcal{F}_h(\hat{r},\hat{s}) \bmod N^2$ by using the trapdoor on the Paillier permutation and knowing p,q, i.e. $(\hat{r},\hat{s}) \leftarrow \mathcal{F}_h^{-1}(\hat{c}(1+mN)^{-1})$. Thus, given a $\hat{c}$, an adversary can find $\hat{d}$ for any message m and thus defeats the binding property.

Oracle Trapdoor Commitment. Finally, we consider trapdoor commitment schemes in which commit, open, and equivocate are given as oracles (and not as algorithms). In such cases, access to equivocate with an input $\hat{c}$ equal to any c which was output by commit is prohibited.

There is a very simple oracle trapdoor commitment scheme in the random oracle model:

- The setup() algorithm is unused.
- The commit(m) oracle with input message m in $\{0,1\}^k$ picks a random value e in $\{0,1\}^\ell$, builds $d \leftarrow (m,e)$, and calls the random oracle $c \leftarrow H(m,e)$.
- The open(c,d) oracle simply extracts m from d and checks that $c = H(m,e)$.
- The equivocate(m,c) oracle yields a decommit value $d = (m,e)$ such that $c = H(m,e)$ by modifying the table of H. This is possible without modifying the final distribution of H, except with probability less than $(Q+C)(2^{-\ell}+2^{-k})$ since c is independent from previous oracle calls.

3 Previous Non-interactive Authentication Protocols

3.1 A NIMAP Based on a Collision-Resistant Hash Functions

We first present a protocol taken from Balfanz et al. [BSSW02] based on a collision resistant hash function.

Note that the authenticated string is constant for all instances of the protocol which use the same input m, i.e. the authenticated string is $H(m)$. This characteristic allows

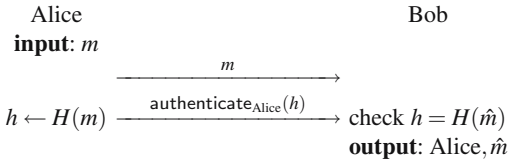


Fig. 4. Non-Interactive Message Authentication using a CRHF

adversaries to run completely offline attacks. An attacker has *simply* to find a collision on the hash function between two messages m_1 and m_2 and then succeeds with probability 1.

Theorem 2 ([Vau05]). *Let μ be the overall time complexity of the message authentication protocol in Fig. 4 using weak authentication. We denote by T, Q, and p the time complexity, number of oracle queries* launch*, and probability of success of adversaries, respectively. There is a generic transformation which transforms any adversary into a collision finder on H whose complexity is $T + \mu Q$ and probability of success is p.*

In short, the best known offline attack against this protocol is the collision attack. An adversary has a probability of success of $1 - e^{-\frac{1}{2}T^2 2^{-k}}$ by using T hashes computations. It clearly succeeds for $T = O(2^{k/2})$. Collision resistance requires the number of authenticated bits to be at least 160 and cannot be reduced considering offline attacks and using only weak authentication.

3.2 A NIMAP with Strong Authentication

The Gehrmann-Mitchell-Nyberg MANA I [GMN04] protocol is depicted in Fig. 5.[1]

MANA I uses a universal hash function family H. Proposed constructions lead to 16–20 bit long SAS values but require strong authentication. Indeed, using weak authentication, an adversary who gets $\mathsf{authenticate}(K||\mu)$ has enough time to find a message $\hat{m}$ such that $\mu = H_K(\hat{m})$ and to substitute m with $\hat{m}$. We can also achieve security with a stronger authenticated channel which achieves stall-free transmissions.

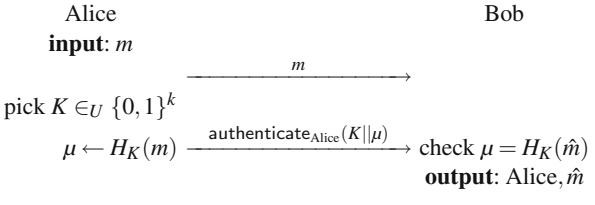


Fig. 5. The MANA I Protocol

[1] Note that the original MANA I protocol is followed by an authenticated acknowledgment from Bob to Alice in [GMN04].

Theorem 3. *Given an ε-universal hash function family H, any adversary which is bounded by a complexity T and by Q_A (resp. Q_B) instances of Alice (resp. Bob) against the protocol of Fig. 5 using stall-free authentication has a probability of success at most $Q_A Q_B \varepsilon$.*

Proof. A one-shot adversary has no advantage to send $\hat{m}$ before it has received m and he cannot send $\hat{m}$ after $K||\mu$ is released. Indeed, he would not be able to send $\hat{m}$ after receiving $K||\mu$ due to the stall-free assumption. Thus, the attacker must select m and $\hat{m}$ and hope that $H_K(\hat{m}) = H_K(m)$. Clearly, the assumption on H limits the probability of success to ε.

Now, consider powerful adversaries. Using Lemma 1, we can deduce that the probability of success of an adversary is at most $Q_A Q_B \varepsilon$. □

4 A Proposed NIMAP with Weak Authentication

Consider the protocol depicted on Fig. 6 in which the message m is transmitted by sending $(c,d) \leftarrow \mathsf{commit}(K_p, m)$. This message can be recovered by anyone using the open function. To authenticate this message, the hashed value of c is sent using an authenticated channel. We prove that this protocol is secure with authenticated strings which can be shorter than in the protocol of Fig. 4. Non-deterministic commitment scheme is the heart of the protocol since an attacker cannot predict the c value and thus cannot predict the $H(c)$ value which is the authenticated one.

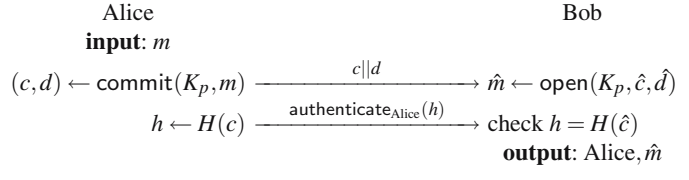


Fig. 6. Non-Interactive Message Authentication Based on a WCRHF

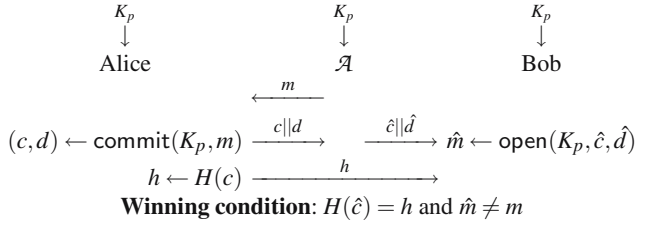
Lemma 4. *Consider the message authentication protocol depicted in Fig. 6. We assume that the function H is a $(T+\mu, \varepsilon_h)$-weakly collision resistant hash function and the commitment scheme is a $(T+\mu, \varepsilon_c)$-trapdoor commitment scheme in the CRS model (resp. oracle commitment scheme). There exists a (small) constant μ such that for any T, any one-shot adversary against this message authentication protocol with complexity bounded by T has a probability of success p smaller than $\varepsilon_h + \varepsilon_c$.*

Recall that the c value is sent through the insecure broadband channel and thus has not to be minimized. Thus, we can use an ε_c as small as desired since we can use any commitment scheme as secure as desired.

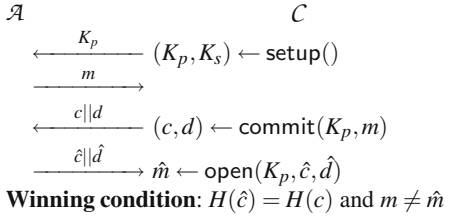
Assuming that H is optimally WCR, the best WCR attack using T hash computations has a probability of success $\varepsilon_h \approx 1 - e^{-T2^{-k}}$. So, we need $T = \Omega(2^k)$ to succeed with a one-shot attack. Thus, using the same amount of authenticated bits as the protocol

of Fig. 4, our protocol has a better resistance against offline attacks. Equivalently, we can achieve the same security as the protocol of Fig. 4, but using only half amount of authenticated bits, e.g. 80 bits.

Proof. A one-shot adversary $\mathcal{A}$ against the protocol in Fig. 6 follows the game depicted on Fig. 7(a) in which it runs a man-in-the middle attack. Clearly, it can be reduced to an adversary $\mathcal{A}$ who plays the game described in Fig. 7(b).



(a)



(b)

Fig. 7. Game Against the Proposed Protocol (a) and Reduced Game (b)

Assume a one-shot adversary $\mathcal{A}$ bounded by a complexity T. Given c, the adversary $\mathcal{A}$ has to find a $\hat{c}$ such that $H(\hat{c}) = H(c)$. In addition, it must find a $\hat{d}$ which opens to $\hat{m}$ (using $\hat{c}$) which is different from the input m. He can of course choose a $\hat{c}$ either equal or either different to c. We study the two cases.

Case 1. ($\hat{c} = c$) The adversary $\mathcal{A}$ chooses $\hat{c}$ equal to c and obviously fulfills the condition $H(\hat{c}) = H(c)$. As depicted on Fig. 8, we can reduce the adversary $\mathcal{A}$ to an adversary against the binding game of Fig. 6. We use an algorithm $\mathcal{B}$ bounded by complexity μ which plays the binding game with a challenger $\mathcal{C}$ on one side and simulates a challenger for $\mathcal{A}$ on the other side at the same time. Using adversary $\mathcal{A}$ and algorithm $\mathcal{B}$, we construct an adversary $\mathcal{AB}$ which plays the binding game. Note that adversary $\mathcal{AB}$ has a complexity bounded by $T + \mu$.

First, the challenger $\mathcal{C}$ generates the pair of keys (K_p, K_S) and sends K_p to $\mathcal{B}$. $\mathcal{B}$ sends it to $\mathcal{A}$ and receives a message m from $\mathcal{A}$. He computes (c,d) using the commit function with K_p and sends $c||d$ to $\mathcal{A}$. As assumed, $\mathcal{A}$ chooses a $\hat{c}$ equal to c and also sends $\hat{c}||\hat{d}$ to $\mathcal{B}$. $\mathcal{B}$ can now deduce $\hat{m}$ using the open function with inputs c and $\hat{d}$. Finally, $\mathcal{B}$ sends all required values to the challenger $\mathcal{C}$.

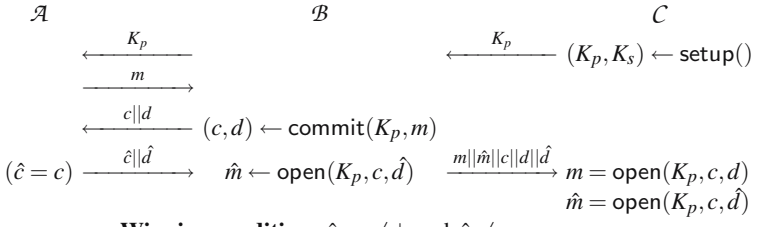


Fig. 8. Reduction to the SB game ($\hat{c} = c$)

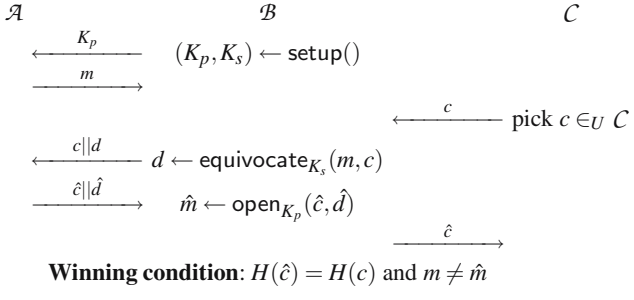


Fig. 9. Reduction to the WCR Game with Trapdoor Commitment ($\hat{c} \neq c$)

Note that $\mathcal{B}$ simulates perfectly a challenger for $\mathcal{A}$. Hence, $\mathcal{A}$ and $\mathcal{AB}$ win their respective game at the same time. Consequently, both win with the same probability of success. Recall that the probability of success of an adversary bounded by a complexity $T + \mu$ against the binding game of Fig. 6 is smaller than ε_c when the commitment scheme is a $(T + \mu, \varepsilon_c)$-trapdoor commitment. Hence, the probability that $\mathcal{A}$ succeeds and $c = \hat{c}$ is at most ε_c. Note that this case equally applies to trapdoor commitment schemes.

Case 2. ($\hat{c} \neq c$) The adversary $\mathcal{A}$ searches a $\hat{c}$ different from c. As depicted on Fig. 9, we can reduce the adversary $\mathcal{A}$ to an adversary against a second preimage search game. We use an algorithm $\mathcal{B}$ bounded by a complexity μ with the help of one query to the $\mathsf{equivocate}$ oracle. $\mathcal{B}$ plays the second preimage game with a challenger $\mathcal{C}$ on one side and simulate a challenger for $\mathcal{A}$ on the other side at the same time. Using adversary $\mathcal{A}$ and algorithm $\mathcal{B}$, we construct an adversary $\mathcal{AB}$ which plays the second preimage game with the challenger $\mathcal{C}$. Note that adversary $\mathcal{AB}$ has a complexity bounded by $T + \mu$.

First, $\mathcal{B}$ generates the keys and sends K_p to $\mathcal{A}$. $\mathcal{B}$ receives a message m from $\mathcal{A}$ and receives a challenge c from $\mathcal{C}$. $\mathcal{B}$ can deduce the decommit value d by calling the oracle $\mathsf{equivocate}(m, c)$. Note that c has been picked uniformly and consequently the distribution of (c, d) is the same as if they have been yield by the commit algorithm. Then, $\mathcal{B}$ can send $c||d$ to $\mathcal{A}$. $\mathcal{A}$ sends a $\hat{c}||\hat{d}$ to $\mathcal{B}$. Finally, $\mathcal{B}$ sends it to the challenger $\mathcal{C}$.

Note that $\mathcal{B}$ simulates perfectly a challenger for $\mathcal{A}$. Hence, $\mathcal{A}$ and $\mathcal{AB}$ win their respective game at the same time and consequently with the same probability of

success. Recall that the probability of success of an adversary against a second preimage game bounded by a complexity $T+\mu$ is smaller than ε_h when H is a $(T+\mu,\varepsilon_h)$-weakly collision-resistant hash function. Hence, the probability that $\mathcal{A}$ succeeds and $c \neq \hat{c}$ is at most ε_h. Note that the proof equally applies to oracle commitment schemes since it is unlikely that the challenge c was output by a commit oracle.

We conclude that any one-shot adversary bounded by a complexity T against the protocol of Fig. 6 has a probability of success smaller than $\varepsilon_c+\varepsilon_h$ when the protocol uses a $(T+\mu,\varepsilon_h)$-weakly collision resistant hash function H and a $(T+\mu,\varepsilon_c)$-trapdoor commitment scheme. □

We consider now powerful adversaries.

Theorem 5. *Consider the message authentication protocol of Fig. 6. We assume that the function H is a $(T+\mu,\varepsilon_h)$-weakly collision resistant hash function and the commitment scheme is a $(T+\mu,\varepsilon_c)$-trapdoor commitment scheme in the CRS model (resp. oracle commitment scheme). There exists a (small) constant μ such that for any T, any adversary against this message authentication protocol with complexity bounded by T and with number of Alice's (resp. Bob's) instances bounded by Q_A (resp. Q_B) has a probability of success p at most $Q_A(\varepsilon_h+\varepsilon_c)$.*

Assuming that WCR hash functions and trapdoor commitments such that $\varepsilon_c \ll \varepsilon_h = O(T2^{-k})$ exist, we have $p = O(T \cdot Q_A 2^{-k})$. As an example, assuming that an adversary is limited to $Q_A \leq 2^{10}$, $T \leq 2^{70}$, and that the security level requires $p \leq 2^{-20}$, the protocol of Fig. 4 requires $k \geq 160$ and our protocol requires $k \geq 100$. Using MD5 [Riv92], our protocol still achieves a quite luxurious security even though collisions have been found on MD5 [WY05].

Proof. Consider an adversary who launches Q_A instances of Alice and Q_B instances of Bob. Clearly, we can simulate all instances of Bob, pick one who will make the attack succeed, and launch only this one. Hence, we reduce to $Q_B = 1$. Recall from Lemma 4 that any one-shot adversary has a probability of success smaller than $\varepsilon_h+\varepsilon_c$. Using Lemma 1, we conclude that any adversary has a probability of success at most $Q_A(\varepsilon_h+\varepsilon_c)$. □

5 On the Required Entropy of Authenticated Communications

Using a weak authenticated channel, adversaries can delay or replay authenticated messages. With non-interactive protocols an adversary can run the catalog attack: i.e. he launches several instances of Alice and recover many authenticated SAS. He launches one Bob and use one SAS of the catalog.

We would like to upper bound the security of an arbitrary message authentication protocol given the amount of authenticated strings it uses. Assume that the protocol is used between Alice and Bob. We suppose that the protocol can use any sequence of authenticated messages in a given set S during the protocol. We call it a *transcript*. Note that authenticated strings are interleaved with regular messages which are not

represented in the transcript. For any input message m, the used transcript during a protocol instance is picked in the set S of all possible transcripts with a distribution $\mathcal{D}_m$.

Theorem 6. *We consider an arbitrary message authentication protocol between Alice and Bob which uses an authenticated channel. Let S be the set of all possible protocol transcripts through the authentication channel for any input message. Let s be its cardinality. There exists a generic one-shot attack with probability of success at least $\frac{1}{s} - 2^{-t}$ which runs in polynomial time in terms of t.*

Proof. We consider a general man-in-the-middle attack in which the adversary first picks $m \in_U \{0,1\}^t$ and $\hat{m} \in_U \{0,1\}^t$ and launches Alice with input m. The attack runs synchronized protocols between Alice and a simulator for Bob, and a simulator for Alice with input $\hat{m}$ and Bob. Following the attack, every authenticated message which must be sent by the simulator is replaced by an authenticated message which has just been received by the simulator.

Let SAS_m be the (random) sequence of all authenticated strings (the transcript) which would be exchanged in the protocol between Alice and the simulator if the simulator where honest, and $SAS_{\hat{m}}$ be the similar sequence between the simulator and Bob. Clearly, if $SAS_{\hat{m}} = SAS_m$, the attack succeeds. Note that an attack makes sense only if $\hat{m}$ is different of m.

We have

$$\begin{aligned}\Pr[\text{success}] &= \Pr[SAS_m = SAS_{\hat{m}} \text{ and } m \neq \hat{m}] \\ &\geq \Pr[SAS_m = SAS_{\hat{m}}] - \Pr[m = \hat{m}].\end{aligned}$$

Note that SAS_m and $SAS_{\hat{m}}$ are two identically distributed independent random variables whose support are included in S. Due to Lemma 8 (see Appendix) we can write $\Pr[SAS_m = SAS_{\hat{m}}] \geq \frac{1}{s}$. Since m and $\hat{m}$ are uniformly distributed in $\{0,1\}^t$, we have $\Pr[m = \hat{m}] = 2^{-t}$. Finally, we obtain

$$\Pr[\text{success}] \geq \frac{1}{s} - 2^{-t}$$

with equality if and only if the SAS distribution is uniform among the set S. □

We finally provide a generic attack in the general case.

Theorem 7. *We consider an arbitrary NIMAP between Alice and Bob which uses a weak authenticated channel. Let S be the set of all possible protocol transcripts through the authentication channel for any input message. Let s be its cardinality. There exists a generic attack which uses Q_A instances of Alice and an offline complexity $O(T)$ with probability of success approximately $1 - e^{-\frac{T \cdot Q_A}{s}}$.*

Proof (Sketch). We consider the generic attack in which the adversary starts by simulating T Alice instances launched with random inputs $\hat{m}_i$ and obtains a list of possible SAS, i.e. $\widehat{SAS}_i$. Then, he launches Q_A real instances of Alice with random inputs m_j and consequently obtains Q_A authenticated SAS, i.e. SAS_j. The attack succeeds when at least one authenticated SAS released by Alice corresponds to a computed one, i.e.

there exists k, ℓ such that $SAS_k = \widehat{SAS}_\ell$. The adversary can launch a single Bob with input $\hat{m}_\ell$ by simulating Alice and can use SAS_k for the authentication when needed.

If the distribution of all SAS is uniform, we have a birthday effect and thus the probability of success is approximately $1 - e^{-\frac{T \cdot Q_A}{s}}$. When the distribution is not uniform, the probability is even larger (see Appendix B of [Pas05]). □

Theorem 6 says that there exists a one-shot attack against *any* message authentication protocol which succeeds with probability essentially $\frac{1}{s}$ where s is the size of S. Theorem 7 says that there exists a generic attack against *any* NIMAP which uses a weak authenticated channel which succeeds with probability essentially $1 - e^{-\frac{T \cdot Q_A}{s}}$ where Q_A is the number of instances of Alice used. Hence, they cannot be secure unless $T \cdot Q_A$ is negligible against s. Thus, any NIMAP which is secure for $T \cdot Q_A \ll s$ is optimal.

Consequently, our proposed protocol is optimal due to Theorem 5 provided that WCR hash functions and trapdoor commitment schemes such that $\varepsilon_c \ll \varepsilon_h = O(T 2^{-k})$ exist. By comparison with our protocol, we can note that the protocol of Fig. 4 is not optimal.

6 Applications

One key issue in cryptography is to setup secure communications over insecure channels, such as Internet. We know that using public key cryptography it is possible by exchanging public keys in an authenticated way. The proposed protocol is used in this case for public key authentications, e.g. GPG public keys. Typical applications where public key cryptography is used, and consequently public key authentication is required, are

- distant hosts authentication, e.g. SSH
- e-mail authentication, e.g. GPG signature
- secure e-mail, e.g. GPG encryption
- secure voice over IP, e.g. PGPfone

Another possible application can be authentication of legal documents. For instance, if two persons would exchange a document without complex appendix, such as GPG signature, they can simply send the corresponding commit and decommit values and then authenticate the hashed commit value. The recipient can check whether or not it is correct. Note that integrity is protected.

7 Conclusion

In this paper, we have proposed a new non-interactive message authentication protocol based on a commitment scheme. It has the same security as the currently used in SSH against one-shot attacks but using only half authenticated bits, e.g. 80 bits. 100 bits only are required against more general attacks. Indeed, due to the commitment scheme, the authenticated value is not foreseeable and the protocol is resistant to collision attacks. The latter theorem proposes that our protocol is optimal. We can in addition conclude on the non-optimality of the protocol used today, but the question about MANA I is still opened. Finally, we stress that the security of our protocol relies essentially on the hardness of the SB game of the commitment scheme and on the hardness on the WCR game of the hash function.

References

[BCC88] Gilles Brassard, David Chaum, and Claude Crépeau. Minimum disclosure proofs of knowledge. *Journal of Computer and System Sciences*, 37(2):156–189, 1988.

[BCJ+05] Eli Biham, Rafi Chen, Antoine Joux, Patrick Carribault, Christophe Lemuet, and William Jalby. Collisions of SHA-0 and reduced SHA-1. In *Advances in Cryptology – EUROCRYPT '05: 24th Annual International Conference on the Theory and Applications of Cryptographic Techniques*, Lecture Notes in Computer Science, pages 36–57, Aarhus, Denmark, 2005. Springer-Verlag.

[BK90] Joan F. Boyar and Stuart A. Kurtz. A discrete logarithm implementation of perfect zero-knowledge blobs. *Journal of Cryptology*, 2(2):63–76, 1990.

[BR93] Mihir Bellare and Phillip Rogaway. Entity authentication and key distribution. In Douglas R. Stinson, editor, *Advances in Cryptology – CRYPTO '93: 13th Annual International Cryptology Conference*, volume 773 of *Lecture Notes in Computer Science*, pages 232–249, Santa Barbara, California, U.S.A., 1993. Springer-Verlag.

[BSSW02] Dirk Balfanz, Diana K. Smetters, Paul Stewart, and H. Chi Wong. Talking to strangers: Authentication in ad-hoc wireless networks. In *Proceedings of Network and Distributed System Security Symposium 2002 (NDSS'02)*, San Diego, California, U.S.A, February 2002.

[CGHGN01] Dario Catalano, Rosario Gennaro, Nick Howgrave-Graham, and Phong Q. Nguyen. Paillier's cryptosystem revisited. In *CCS '01: Proceedings of the 8th ACM conference on Computer and Communications Security*, pages 206–214, Philadelphia, Pennsylvania, U.S.A., 2001. ACM Press.

[DG03] Ivan Damgård and Jens Groth. Non-interactive and reusable non-malleable commitment schemes. In *STOC '03: Proceedings of the thirty-fifth annual ACM symposium on Theory of computing*, pages 426–437, San Diego, California, U.S.A., 2003. ACM Press.

[GMN04] Christian Gehrmann, Chris J. Mitchell, and Kaisa Nyberg. Manual authentication for wireless devices. *RSA Cryptobytes*, 7(1):29–37, January 2004.

[LdW05] Arjen K. Lenstra and Benne de Weger. On the possibility of constructing meaningful hash collisions for public keys. In Colin Boyd and Juan Manuel González Nieto, editors, *ACISP '05: The 10th Australasian Conference on Information Security and Privacy*, volume 3574 of *Lecture Notes in Computer Science*, pages 267–279, Brisbane, Australia, 2005. Springer-Verlag.

[LWdW05] Arjen Lenstra, Xiaoyun Wang, and Benne de Weger. Colliding X.509 certificates. Cryptology ePrint Archive, Report 2005/067, 2005. `http://eprint.iacr.org/`.

[Pai99] Pascal Paillier. Public-key cryptosystems based on composite degree residuosity classes. In Jacques Stern, editor, *Advances in Cryptology – EUROCRYPT '99: International Conference on the Theory and Application of Cryptographic Techniques*, volume 1592 of *Lecture Notes in Computer Science*, pages 223–238, Prague, Czech Republic, May 1999. Springer.

[Pas05] Sylvain Pasini. Secure communications over insecure channels using an authenticated channel. Master's thesis, Swiss Federal Institute of Technology (EPFL), 2005. `http://lasecwww.epfl.ch/php_code/publications/search.php?ref=Pas05`.

[Riv92] Ronald L. Rivest. The MD5 message digest algorithm. Technical Report Internet RFC-1321,IETF, 1992.

[Vau05] Serge Vaudenay. Secure communications over insecure channels based on short authenticated strings. In Victor Shoup, editor, *Advances in Cryptology – CRYPTO '05: The 25th Annual International Cryptology Conference*, volume 3621 of *Lecture Notes in Computer Science*, pages 309–326, Santa Barbara, California, U.S.A., August 2005. Springer-Verlag.

[WLF+05] Xiaoyun Wang, Xuejia Lai, Dengguo Feng, Hui Chen, and Xiuyuan Yu. Cryptanalysis of the hash functions MD4 and RIPEMD. In Ronald Cramer, editor, *Advances in Cryptology – EUROCRYPT '05: 24th Annual International Conference on the Theory and Applications of Cryptographic Techniques*, Lecture Notes in Computer Science, pages 1–18, Aarhus, Denmark, 2005. Springer-Verlag.

[WY05] Xiaoyun Wang and Hongbo Yu. How to break MD5 and other hash functions. In Ronald Cramer, editor, *Advances in Cryptology – EUROCRYPT '05: 24th Annual International Conference on the Theory and Applications of Cryptographic Techniques*, Lecture Notes in Computer Science, pages 19–35, Aarhus, Denmark, 2005. Springer-Verlag.

[WYY05a] Xiaoyun Wang, Yiqun Yin, and Hongbo Yu. Finding collisions in the full SHA1. In Victor Shoup, editor, *Advances in Cryptology – CRYPTO '05: The 25th Annual International Cryptology Conference*, volume 3621 of *Lecture Notes in Computer Science*, pages 17–36, Santa Barbara, California, U.S.A., 2005. Springer-Verlag.

[WYY05b] Xiaoyun Wang, Xiuyuan Yu, and L. Y. Yin. Efficient collision search attacks on SHA-0. In Victor Shoup, editor, *Advances in Cryptology – CRYPTO '05: The 25th Annual International Cryptology Conference*, volume 3621 of *Lecture Notes in Computer Science*, pages 1–16, Santa Barbara, California, U.S.A., 2005. Springer-Verlag.

Appendix

Lemma 8. *Let X and Y be two identically distributed independent random variables with distribution D over a support set S. We have*

$$\Pr[X = Y] \geq \frac{1}{\#S} \tag{1}$$

with equality if and only if D is the uniform distribution.

Proof. Let s be the size of the set S. We have

$$\Pr[X = Y] = \sum_{S_i \in S} \Pr[X = S_i] \cdot \Pr[Y = S_i] = \sum_{S_i \in S} p_i^2$$

where p_i is $\Pr[X = S_i]$.

Let us write $p_i = \frac{1}{s} + \rho_i$. Thus, we obtains

$$\sum_{S_i \in S} p_i^2 = (\frac{1}{s})^2 \sum_{S_i \in S} 1 + 2\frac{1}{s} \sum_{S_i \in S} \rho_i + \sum_{S_i \in S} \rho_i^2.$$

Knowing that the sum of p_i equals to 1, we can easily deduce that the sum of ρ_i equals 0. Thus, $\sum_{S_i \in S} p_i^2$ equals $\frac{1}{s} + \sum_{S_i \in S} \rho_i^2$. The sum of ρ_i^2 is greater or equal to 0. Note that it is equal to 0 if and only if all ρ_i are null, i.e. D is uniform. □