NONINTERACTIVE TWO-CHANNEL MESSAGE AUTHENTICATION BASED ON HYBRID-COLLISION RESISTANT HASH FUNCTIONS

ATEFEH MASHATAN[1] AND DOUGLAS R. STINSON[2]

Abstract. We consider the problem of non-interactive message authentication using two channels: an insecure broadband channel and an authenticated narrow-band channel. This problem has been considered in the context of ad hoc networks, where it is assumed that there is neither a secret key shared among the two parties, nor a public-key infrastructure in place. We present a formal model for protocols of this type, along with a new protocol which is as efficient as the best previous protocols. The security of our protocol is based on a new property of hash functions that we introduce, which we name "hybrid-collision resistance".

1. Introduction

The problem of authentication is of fundamental importance in cryptography. Entity authentication and message authentication are two important aspects of secure communication. Typically, communicating parties would like to be assured of the authenticity of information they obtain via potentially insecure channels, as well as the identity of the sender.

There are many approaches to achieving these goals in standard models of public-key cryptography and secret-key cryptography. However, in ad hoc networks, traditional settings for cryptography may not be appropriate, for various reasons. For example, a public-key infrastructure may not exist; secure channels might not be present; communication bandwidth may be severely limited, etc.

The model we consider is described in detail in [GN04] and [GMN04]. Two small devices wish to establish a secure key in an environment where no public-key infrastructure exists. The two devices can communicate over an insecure broadband

August 2006

[1] Department of Combinatorics and Optimization
University of Waterloo
Waterloo, Ontario CANADA N2L 3G1
amashatan@uwaterloo.ca
[2] David R. Cheriton School of Computer Science
University of Waterloo
Waterloo, Ontario CANADA N2L 3G1
dstinson@uwaterloo.ca .

network. Also available is an authenticated narrow-band channel. This channel might be based on information transmitted by human beings, e.g., a short string that is read from one device and copied to the other device. The short string is used to help to authenticate the information sent over the wide-band channel. In this paper, we concentrate on non-interactive protocols of this type, which are termed NIMAPs (this is an abbreviation for "noninteractive message authentication protocol".)

Balfanz et al [BSSW02] were the first to propose an NIMAP. In their protocol, a message M is transmitted over the broadband channel, and the message digest $H(M)$ is transmitted over the narrow-band channel, where H is a secure hash function. In order to prevent collision attacks, the massage digest should be at least 160 bits in length. In the situation where the narrow-band channel is human operated, however, it might be desirable to reduce the amount of information that has to be sent over this channel, say to 100 bits or even fewer.

Gehrmann, Mitchell and Nyberg [GMN04] proposed several protocols which they called MANA I, MANA II, etc. Their protocols reduce the amount of authenticated information that needs to be sent, but they require a stall-free channel (see [Vau05], for example, for further discussion).

Pasini and Vaudenay [PV06] proposed an NIMAP based on second-preimage resistant hash functions and commitment schemes in the Common Reference String (CRS) model. The CRS model assumes a random string K_p has previously been distributed to all participants in the protocol. In [PV06], two commitment schemes are proposed: an oracle commitment scheme and a trapdoor commitment scheme. Two trapdoor commitment schemes they considered are (1) a scheme proposed by Boyar and Kuntz [BK90], which is based on the discrete logarithm problem, and (2) a scheme proposed by Catalano et al [CGHGN01] based on Paillier's trapdoor permutation [Pai99]. In the schemes proposed in [PV06], the key K_p is of size $N^2 + N$, where the message has size N. Thus K_p could be a rather long key, which must be authenticated in a manner similar to a public key. Furthermore, the commitment schemes have a somewhat complicated structure, especially when compared to other NIMAPs that just use hash functions, for example.

Another recent paper, by Naor, Segev and Smith [NSS], investigates two-channel authentication in the interactive setting. Their protocols are unconditionally secure, but the number of rounds required depends on the the length of the message to be authenticated.

1.1. **Our contributions.** We describe a formal model for NIMAPs using two channels, and analyze the attacks that can occur in this model. Our model allows offline attacks by an adversary, as well as replay attacks. This is a strong attack model,

so a scheme that is proven secure in this model does not require authenticated channels that have any unusual properties.

We show that it is sufficient to consider only impersonation attacks in this model. Security of NIMAPs can be reduced to a certain "binding game". This makes it quite straightforward to analyze protocols in this model.

In preparation for the description of our protocol, we introduce the idea of "hybrid-collision resistant" (HCR) hash functions. After analyzing the HCR property in the random oracle setting, we construct a new NIMAP based on HCR hash functions. Our protocol has a very simple structure and does not require any long strings to be authenticated ahead of time. These properties make the protocol applicable in wide variety of settings. We analyze the security and efficiency of our protocol and compare it to other protocols.

The rest of this paper is organized as follows. Sections 2 and 3 deal with the General Model for NIMAPs over two channels; Section 4 examines previously proposed NIMAPs; and Section 5 proposes a new NIMAP.

In Section 2, a general NIMAP using two channels, GNIMAP, is proposed. The GNIMAP provides the required formalism for NIMAPs over two channels. The attack model, i.e. adversarial goal and capabilities, are defined in Section 3. Further, a Binding Game is introduced and analyzed. Then, GNIMAP is proven to be secure given that the Binding Game is hard to win.

Section 4 is devoted to briefly examine the previous NIMAPs in the literature. The security of three NIMAPs in our General Model is analyzed. Further, the amount information sent in order to achieve a certain level of security is noted.

In Section 5, we define Hybrid-Collision Resistance (HCR) for hash functions. The HCR Game is introduced and is analyzed in order to better understand the hardness of finding Hybrid-Collisions. Moreover, an NIMAP, based on HCR hash functions, is proposed. We prove that our NIMAP is secure given that the HCR Game is hard to win. Furthermore, the simplicity of the structure and the amount of information sent over both channels is compared between our proposed NIMAP and the most secure NIMAP found in the literature.

Finally, Section 6 contains some concluding remarks.

2. General Model

Assume that two channels are accessible for communication: an insecure broadband channel, denoted by $\rightarrow$, and an authenticated narrow-band channel, denoted by $\Rightarrow$. Communication over the narrow-band channel is usually more expensive and less accessible. Hence, the messages sent over the authenticated channel are ideally much shorter than those sent over the insecure channel.

We assume that the adversary cannot modify the information transmitted over the authenticated channel, i.e., data integrity is insured in this channel. Moreover, these narrow-band channels are equipped with authenticating features such that the recipient of the information can be sure about who sent it. However, the adversary can replay a previous flow or remove it.

Now consider a non-interactive Message Authentication Protocol that employs both the authenticated and the insecure channel between a claimant Alice and a verifier Bob. All flows are initiated from Alice and there are a total of two flows, one over the insecure channel and the other over the authenticated channel. We note that there is no flow being initiated from Bob and as a result, the order in which these two flows are being sent over the channels does not matter. Moreover, all other scenarios of a non-interactive Message Authentication Protocol involving more than two flows can be reduced to this scenario. That is, we can simply combine the flows sent over each type of channel in a single flow. This is not the case in the interactive setting since the data sent by Alice may depend on some data sent by Bob in a previous flow, which makes both the order and number of flows important in analysis.

Let $\mathcal{M}$ be the space of messages. In a Message Authentication Protocol, the claimant Alice chooses a message $M \in \mathcal{M}$ and sends it to Bob using the protocol. At the end, Bob either outputs (Alice, M'), where $M' \in \mathcal{M}$, or he rejects.

Consider a randomized algorithm $split : \mathcal{M} \to \mathcal{M}_1 \times \mathcal{M}_2$ which takes any message M as input and maps it into a pair (m_1, m_2), where m_1 is shorter than m_2. The reverse procedure is carried out by a deterministic function $reconstruct : \mathcal{M}_1 \times \mathcal{M}_2 \to \mathcal{M} \cup \{\perp\}$ which takes a pair (m_1, m_2) and maps it into a message $M \in \mathcal{M}$ or a "reject" sign $\perp$.

In order to employ the *split* and *reconstruct* functions in a Message Authentication Protocol, we need them to satisfy the following requirements:

(i) Correctness property: Any message can be uniquely recovered. That is, for any $M \in \mathcal{M}$,
$$reconstruct(split(M)) = M.$$

(ii) Binding property: The Binding game of Figure 1 is hard. In other words, it is computationally infeasible to find a message M such that given (m_1, m_2), where $split(M) = (m_1, m_2)$, one can efficiently find an $m_2' \in \mathcal{M}_2 \setminus \{m_2\}$ so that
$$reconstruct(m_1, m_2') \in \mathcal{M}$$
with non-negligible probability.

Given a pair (m_1, m_2) corresponding to a message M, it is desirable that for all m_2' either $reconstruct(m_1, m_2') = M$ or $reconstruct(m_1, m_2') = \perp$ with high

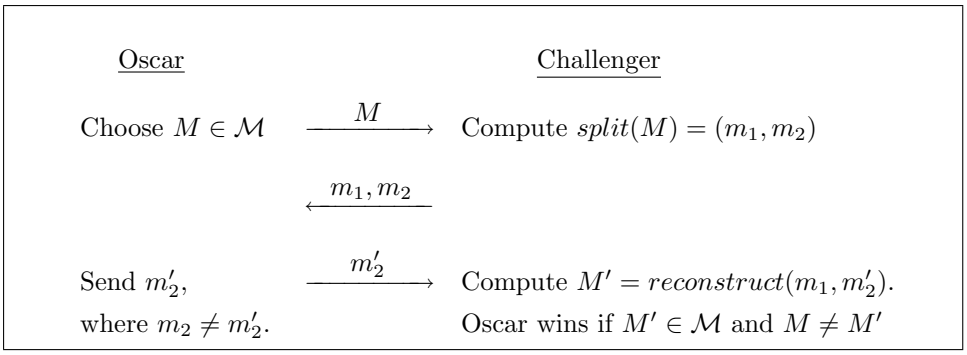

FIGURE 1. The Binding Game

probability. The Binding property insures that the values m_1 and m_2 are bound in such a way that for almost all values of m_2', the pair (m_1, m_2') corresponds to the same message M or it is going to be rejected.

We call a pair of functions $(split, reconstruct)$ to be (T, ϵ)-binding, if any adversary bounded by a complexity T wins the Binding game with a probability of success at most ϵ.

Now consider the following general non-interactive Message Authentication Protocol, where the *split* and *reconstruct* functions satisfy the correctness property and are (T, ϵ)-binding. This protocol, abbreviated as GNIMAP, is also depicted in Figure 2.

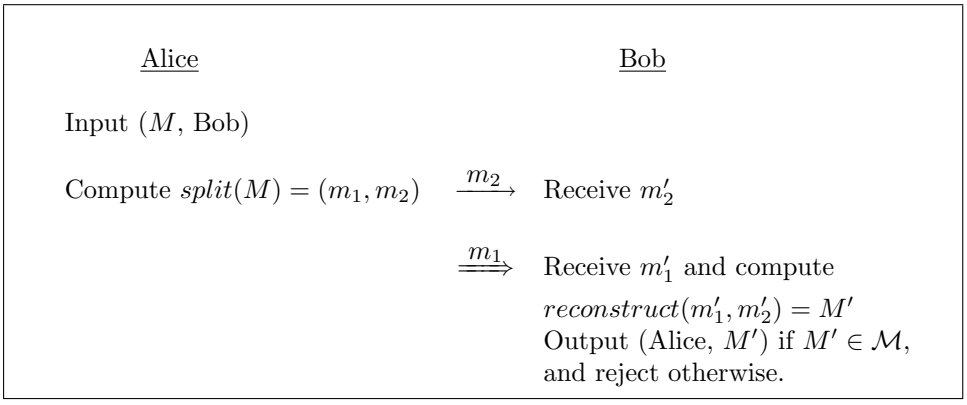

FIGURE 2. General Non-Interactive Message Authentication Protocol

General Non-Interactive Message Authentication Protocol (GNIMAP):

1. On input (M, Bob), Alice computes $split(M) = (m_1, m_2)$.
2. Alice sends m_2 to Bob over the broadband channel.

3. Bob receives m_2'.[1]
4. Alice sends m_1 to Bob over the authenticated channel.
5. Bob receives m_1'.
6. Bob computes $reconstruct(m_1', m_2') = M'$.
7. Bob outputs (Alice, M') if $M' \in \mathcal{M}$, and rejects otherwise.

3. Analysis of the General Model

The correctness of the aforementioned GNIMAP is ensured by property (i). In other words, Bob can successfully recover M from the protocol if all the participants have been honest and no attack has occurred. In order to analyze the security of GNIMAP, we need to define an attack model. Adversarial goal and capabilities are described in the following section.

3.1. **Attack Model.** In the setting of message authentication protocols, the *adversarial goal* is to make Bob accept a message M along with the identity of Alice, when he was supposed to reject (that is, when the message M was never sent by Alice to Bob.) There are two main types of attacks to consider: *impersonation* attacks and *substitution* attacks.

In an impersonation attack, the attacker tries to convince Bob that a message M is sent from Alice, while in fact M was never sent from Alice and the session has been initiated by the adversary. Figure 3 depicts the impersonation attack in the setting of GNIMAP.

Note that, according to our model, the adversary cannot modify the data sent over the authenticated channel, but he or she can replay them. Hence, the authenticated flow in an impersonation attack is replay of a previous flow sent by Alice.

Eve		Bob
Choose m_2'	$\xrightarrow{m_2'}$	
Let $m_1' = m_1$, where Alice has sent m_1 in a previous flow	$\xrightarrow{m_1'}$	Compute $M' = reconstruct(m_1', m_2')$. If $M' \in \mathcal{M}$, then output (Alice, M'), reject otherwise.

Figure 3. An Impersonation Attack Against GNIMAP

[1]Note that, the values that Bob receives might have been altered by an adversary. Hence, we use D' in the receiving end where the data D is transmitted.

In a substitution attack, on the other hand, Alice initiates a session with Bob trying to send him a message M. The adversary then substitutes M' instead of M. So, Bob receives M' and not M. The adversary may have changed part or all of M to get M'. In case of our protocol, the adversary replaces m_2 with m_2', after Alice splits M into (m_1, m_2). The authenticated value m_1 cannot be substituted according to the model.

Note that, the message M might have been chosen by the adversary. In other words, the adversary can make Alice send a message that the adversary has chosen. This ability of the adversary may not be considered in all models. We do consider it in our model since it makes the adversary stronger and results in a stronger model. Figure 4 illustrates a substitution attack against GNIMAP.

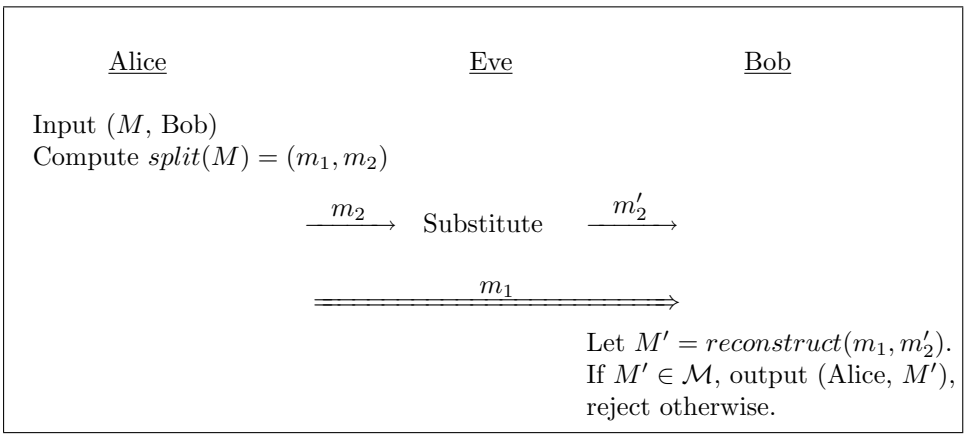

FIGURE 4. A Substitution Attack Against GNIMAP

We consider an adaptive chosen plain-text attack (ACPA) model in our general setting. Note that ACPA model is very strong and desirable compared to other models. An adaptive chosen plain-text attack consists of two stages: an *information gathering* stage and a *deception* stage.

The model presumes that in the information gathering stage, the attacker has the capability to adaptively choose a number of arbitrary messages M_i, and have Alice send them to Bob. The attacker then records the communication for further use. He or she can choose the subsequent messages to be sent by Alice using the results of the messages already sent. The goal of this stage is to gradually reveal information about the unknown aspects of the system (e.g. the randomized *split* function in our case.) In addition, we assume that the attacker has precomputing capabilities and is able to mount "dictionary"-type attacks. The information gathering stage of an attack against GNIMAP is depicted in Figure 5.

Let $\mathcal{N}$ denote the set of all messages M sent by Alice to Bob before the start of deception stage, and the set N denote the set of ordered pairs (m_1, m_2) sent by

Alice to Bob over the two channels before the start of deception stage. Note that, the set $\mathcal{N}$ includes all messages previously sent by Alice to Bob with or without the request of the attacker.

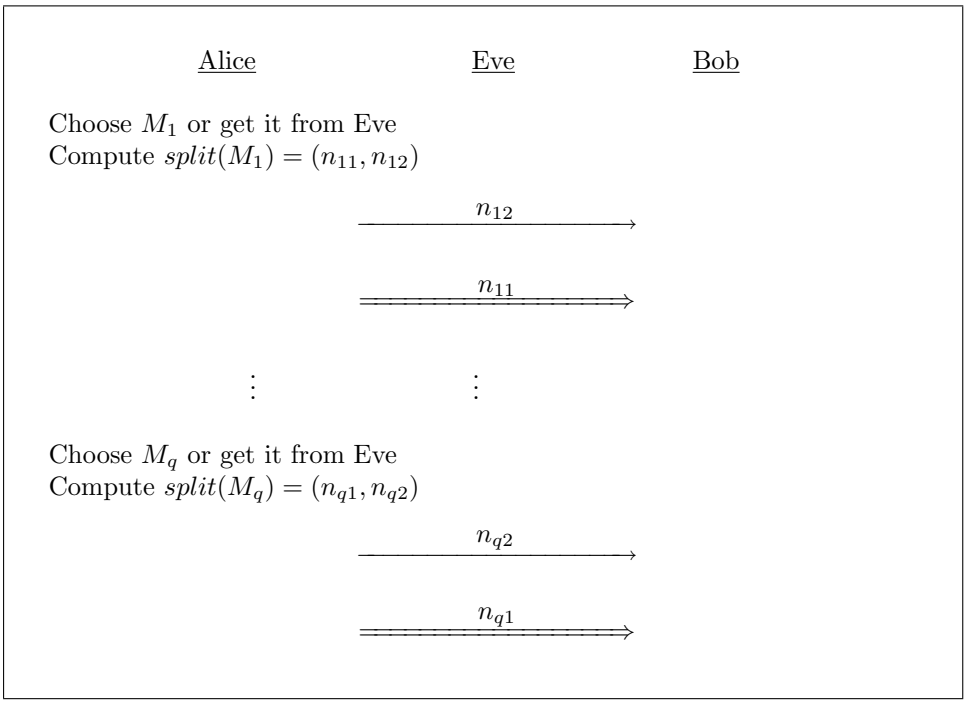

FIGURE 5. Information Gathering Phase of an Attack

We use the term *online complexity* of an adversary to refer to the number q of messages sent by Alice to Bob during the information gathering stage. On the other hand, the term *offline complexity* is used to refer to the computational complexity T of an adversary.

The deception stage is were the attack occurs. That is, the adversary tries to achieve his or her goal by making Bob accept a message M along with the identity of Alice, when he was supposed to reject. The attack is either a substitution or an impersonation attack.

In case of a substitution attack, Alice is sending a pair (m_1, m_2) to Bob. The adversary substitutes m_2 with m_2' and leaves m_1 untouched. Now let M be one of the messages sent by Alice in the information gathering stage. On the other hand, consider an impersonation attack were the adversary sends m_2' and replays m_1. Given that $M \in \mathcal{N}$, this impersonation attack is equivalent to the substitution attack that we started with. This fact is illustrated in Figure 6. Hence, without loss of generality, we only consider impersonation attacks in the deception phase.

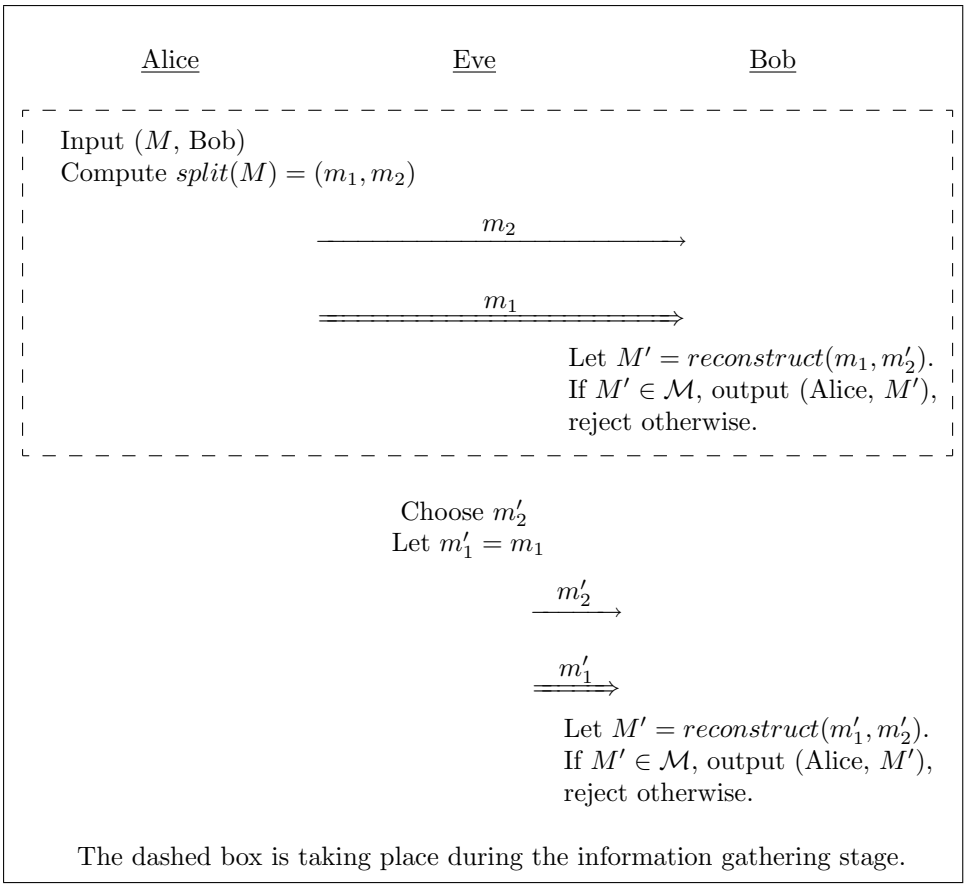

FIGURE 6. Equivalence of Impersonation and Substitution Attacks Against GNIMAP in the ACPA Model.

In the deception stage, the attacker tries to impersonate Alice by sending a single message $M' \notin \mathcal{N}$. The attack succeeds if Bob accepts, and it fails otherwise. In choosing M' the attacker can use all the information obtained from the information gathering stage, which includes the messages sent previously by Alice without the attacker's request. The deception stage is illustrated in Figure 7.

Note that anyone can replay both flows of a previous conversation between Alice and Bob. In this case, Bob accepts a message that was previously sent by Alice. However, this replay impersonation does not constitute an attack. In a successful attack, the adversary is required to replay the second flow and change the first flow. The first flow could be a replay of a previously transmitted first flow. However, the two flows of the attack should not be identical to a previous conversation of Alice and Bob, otherwise the "attack" is considered a replay.

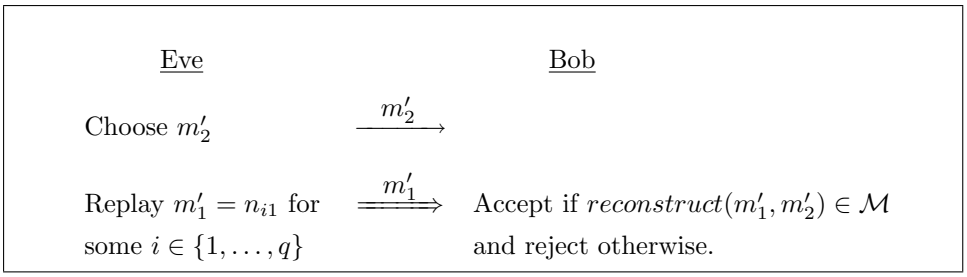

FIGURE 7. Deception Phase of an Attack

3.2. **Security.** In this Section, we prove that GNIMAP is secure given the properties enumerated in Section 2 and under the attack model described in Section 3.1. The proof is based on a reduction.

Associated to each attack, there are sets $\mathcal{N}$ and N, resulting from the information gathering stage, and a pair (m_1', m_2'), from the deception stage, according to our attack model. Let $\mathcal{N} = \{M_1, M_2, \ldots, M_q\}$. Then, $N = \{(n_{i1}, n_{i2}) : 1 \leq i \leq q\} \subset \mathcal{M}_1 \times \mathcal{M}_2$, where $construct(n_{i1}, n_{i2}) = M_i$ for each $1 \leq i \leq q$. The pair (m_1', m_2') corresponds to the deception stage, where the adversary replays m_1' over the authenticated channel, and sends m_2' over the insecure channel.

Let us assume that an attack has occurred and Bob has accepted. That is, the adversary has impersonated Alice by sending the pair (m_1', m_2') to Bob. Moreover, Bob has accepted and has output (M', Alice), where $M' = reconstruct(m_1', m_2')$.

In any successful attack, the adversary needs to replay the authenticated flow. As a result, $m_1' \in \{n_{11}, n_{21}, \ldots, n_{q1}\}$. That is $m_1' = n_{i1}$, for some $1 \leq i \leq q$. Without loss of generality, assume that i is the smallest index for which $m_1' = n_{i1}$. Moreover, $M' \notin \{M_1, M_2, \ldots, M_q\}$, since otherwise the attack is only a replay and not a real attack.

We now formally prove that the GNIMAP is secure given that $(split, reconstruct)$ is (T, ϵ)-binding. That is, we reduce an adversary who can attack the GNIMAP with non-negligible probability to an adversary who wins the Binding game with non-negligible probability.

Consider the game depicted in Figure 8. We call this game the "GNIMAP Game". This is because, if Eve wins this game with probability ϵ, then the game translates into an attack against GNIMAP with success probability ϵ. Here, Eve is facing a challenger who is simulating both Alice and Bob. The game consists of q rounds of Eve sending messages M_i and the challenger responding with (n_{i1}, n_{i2}), where $Split(M_i) = (n_{i1}, n_{i2})$. These q rounds correspond the information gathering phase of the attack. The last round is analogous to the deception phase where Eve, sends her pair (m_1', m_2'). Eve wins the game if $m_1' = n_{i1}$, for some $i \in \{1, \ldots, q\}$, while $reconstruct(m_1', m_2') = M' \neq M_i$.

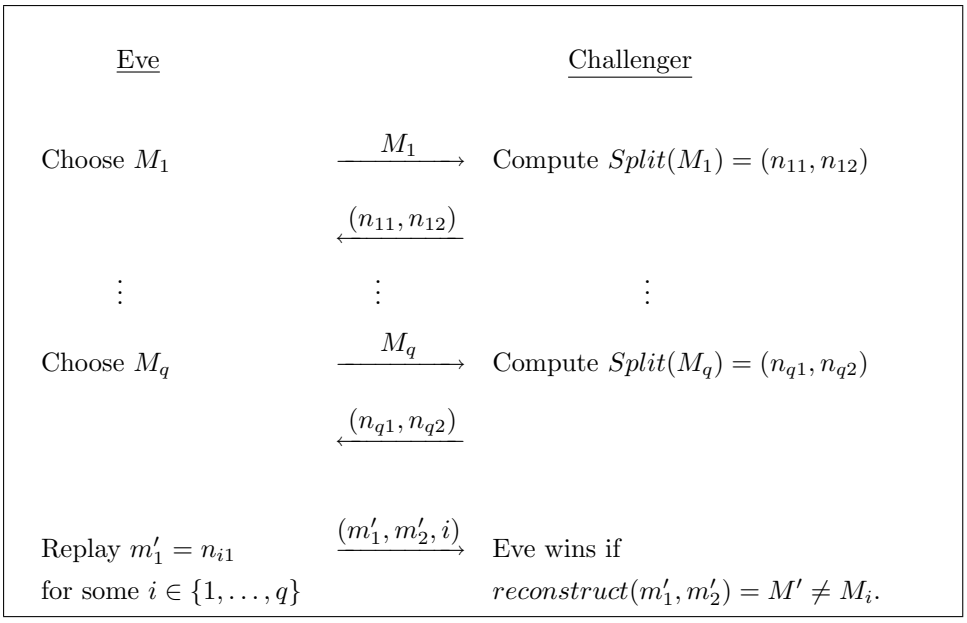

FIGURE 8. GNIMAP Game

Assuming that Eve wins this game with non-negligible probability, we can employ her in the Binding game of Figure 1.

Depicted in Figure 9, Eve is playing against her GNIMAP Game Challenger, while Oscar is playing against his Binding Game Challenger. Oscar will use the results of the GNIMAP Game to win his Binding Game. He first chooses a random value $j \in_R \{1, \ldots, q\}$. Then, Eve will carry out her own attack against the GNIMAP Challenger. That is, Eve sends messages M_t and receives n_{t1} and n_{t2}.

The responses, n_{t1} and n_{t2}, come from computing $split(M_t)$, except when $t = j$. In the jth round, Oscar forwards $M = M_j$ to his challenger. The challenger responds with a pair (m_1, m_2). Then, Oscar forwards $n_{j1} = m_1$ and $n_{j2} = m_2$ to the GNIMAP Challenger.

After q rounds, Eve chooses a message M' and sends m'_1 and m'_2. Note that, for Eve to win, $m'_1 = n_{i1}$ for some $i \in \{1, \ldots, q\}$. Oscar simply forwards m'_2 to his challenger if $j = i$, and quits otherwise.

Note that from Eve's point of view, this game is no different than the game of Figure 8.

Assuming that Eve wins her game with probability ϵ, Oscar clearly wins his game with probability ϵ/q. Hence, we have proved the following Theorem.

Theorem 1. *Assume that there is a GNIMAP where the pair* $(split, reconstruct)$ *is* (T, ϵ)*-binding. In the ACPA model, any adversary against this GNIMAP with*

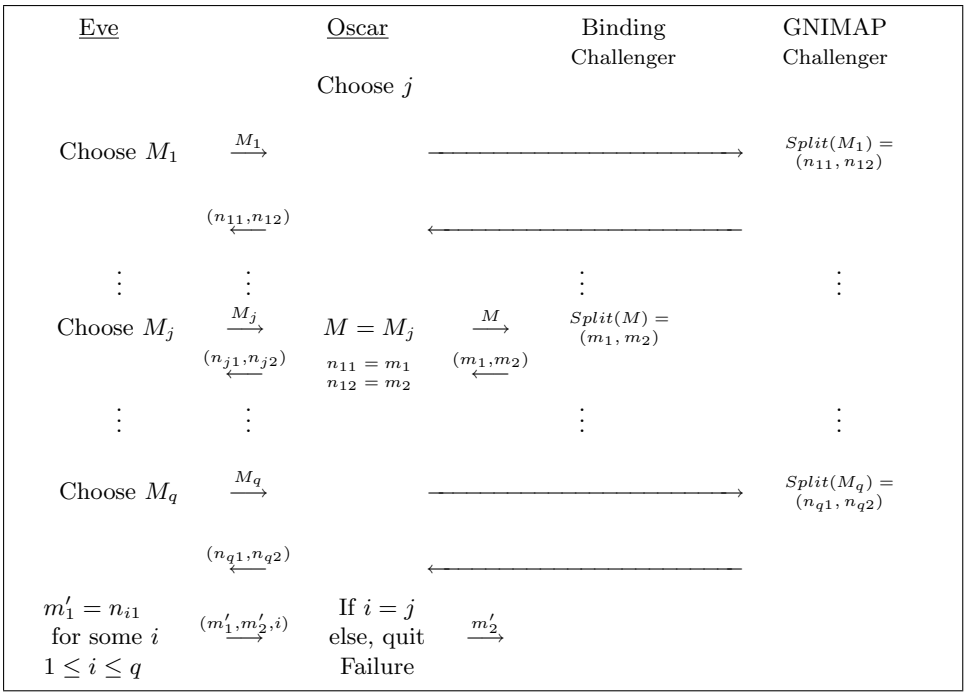

FIGURE 9. Reducing the GNIMAP Game to the Binding Game

online complexity q and offline complexity T has a probability of success p at most $q\epsilon$.

We note that our reduction is not tight. However, it is safe to assume that $q \leq 2^{10}$ in Manual Authentication.[2]

4. Previous Non-interactive Message Authentication Protocols

In this Section, we first define the kind of hash functions that are going to come up in our discussion. Secondly, we briefly introduce the previous NIMAPs found in the literature. Then, the security of these protocols is analyzed with respect to our general model.

4.1. Definitions.

We use the following definitions of different types of Hash functions in the rest of the paper.

A **Collision Resistant Hash Functions, (CR)** H, is a hash function where it is hard to find distinct elements x and y such that $H(x) = H(y)$. The pair (x, y) is called a collision pair. For security purposes, the length of the hash value is required to be more than 160 bits. Otherwise, an adversary has a good chance of finding a collision pair using an offline birthday attack.

[2]The reduction in [PV06] is also not tight and they get the same probability of success, p/q. They also assume that $q \leq 2^{10}$.

A **Second-Preimage Resistant Hash Function, (SPR)** H, is a hash function where given a value x, it is hard to find a value y, $x \neq y$, such that $H(x) = H(y)$. In this case, the best known generic attack is the exhaustive search. Hence, the length of the hash value is required to be at least 80 bits.

An **ϵ-Universal Hash Function Family, (ϵ-UHFF)** H is a collection of functions H_K depending on a random key K, where $\Pr[H_K(x) = H_K(y)] \leq \epsilon$ for any two distinct values x and y.

We now briefly introduce three NIMAPs found in the literature.

4.2. **Balfanz-Smetters-Stewart-Wong NIMAP.** Balfanz et al introduced the idea of hashing the data to be authenticated and delivering the hash value in an authenticated way to the verifier [BSSW02]. Their protocol is based on a collision resistant hash function. It is depicted in Figure 10.

Alice		Bob
Input M	$\xrightarrow{M}$	Receive M'
Compute $h = H(M)$	$\xRightarrow{h}$	Receive h' and accept if $h' = H(M')$. Reject otherwise.

FIGURE 10. Balfanz et al NIMAP

The adversary can work offline and find a collision M_1 and M_2 yielding the same hash value. Then, M_1 is given to Alice in the information gathering stage and she sends Bob the value of $H(M_1)$ over the authenticated channel. The adversary replays this authenticated flow along with M_2 and makes Bob accept. This attack is depicted in Figure 11. If the adversary can mount the above attack efficiently, then this protocol fails to satisfy property (ii) of Section 2.

The collision pair, M_1 and M_2, could be found using a "birthday attack". Birthday attacks have square root complexity. If we consider algorithms of complexity 2^{80} inefficient, then in order to make this attack not efficient we need to increase the size of the authenticated bits, i.e. h, to 160 bits.

4.3. **Gehrmann-Mitchell-Nyberg NIMAP: MANA I.** Gehrmann et al introduced MANA I based on an ϵ-universal hash function family H [GMN04]. This protocol is depicted in Figure 12. In their original proposal, confidentiality of the authenticated channel is required. This requirement is very restrictive in general. In [Vau05], Vaudenay has proved that a "stall-free" authenticated channel is enough to ensure the security of MANA I. However, the stall-free requirement is still quite strong and not desirable in an arbitrary authenticated channel.

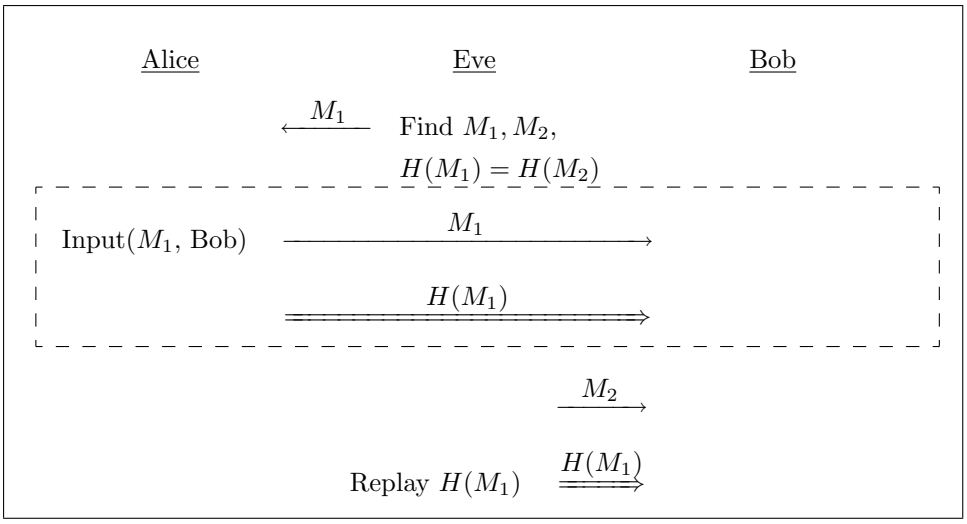

FIGURE 11. Attack against the Balfanz et al NIMAP

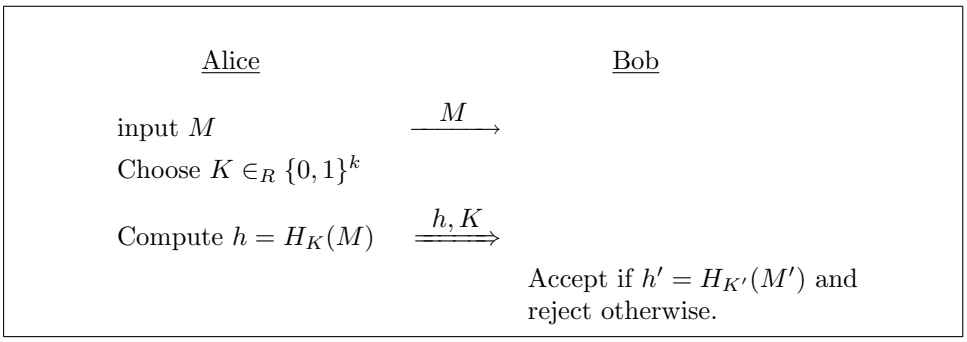

FIGURE 12. MANA I

The use of universal hash families makes MANA I not secure in our model. The adversary records a pair $(H_K(M), K)$ from the information gathering stage and finds M' such that $H_K(M) = H_K(M')$. This is usually an easy computation since the function H_K is not required to be collision free. He or she then sends M' over the insecure channel and replays $(H_K(M), K)$ over the authenticated channel.

4.4. **Pasini-Vaudenay NIMAP.** Pasini and Vaudenay proposed a NIMAP, illustrated in Figure 13, based on Second-Preimage Resistant hash functions [PV06]. The protocol is in the Common Reference String (CRS) model, which assumes a random string K_p has been previously distributed to everyone. The *commit* function has two inputs: the message M and the CRS K_p. It outputs a commit value c and a decommit value d. This function is non-deterministic and is playing the role of the *split* function. The *open* function, on the other hand, is a deterministic function. It uniquely outputs M on input (K_p, c, d).

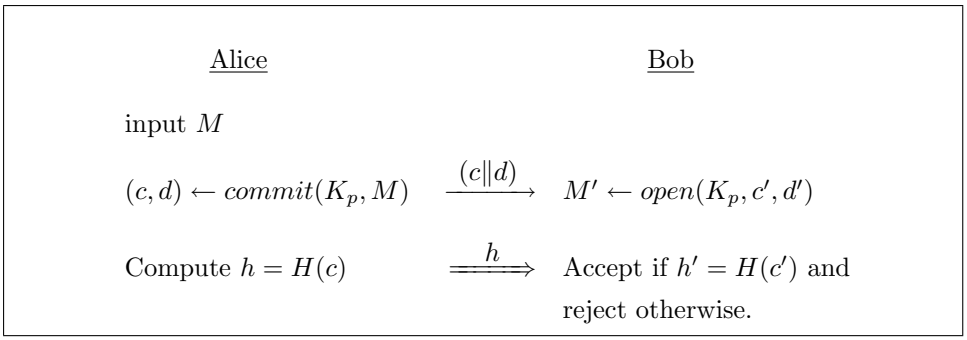

FIGURE 13. Pasini-Vaudenay NIMAP

In [PV06], an adversary attacking the NIMAP is reduced to an adversary who finds second-preimages or breaks the trapdoor of the commitments. To achieve security against an adversary with online complexity of 2^{80} and $q = 2^{10}$, they need to authenticate 100 bits. More details can be found in [PV06].

There is always the issue of authenticity attached to public parameters such as K_p. Hence, it possibly restricts the application of this NIMAP. Moreover, as discussed in the Introduction, we are trying to replace the use of any PKI by using NIMAPs. As a result, this protocol does not seem to be the optimal solution.

On the other hand, this NIMAP is based on the assumption that trapdoor commitment schemes exist, as well as SPR hash functions. This protocol satisfies the properties of Section 2.

5. A non-interactive Message Authentication Protocol using Hybrid-Collision Resistant Hash Functions

In this Section, we first define Hybrid-Collision Resistance for hash functions. Secondly, we discuss the difficulty of finding hybrid-collisions. Moreover, a new NIMAP based on Hybrid-Collision Resistant hash functions is introduced. The security of this NIMAP is ensured by showing that it satisfies the properties we listed in Section 2 when using Hybrid-Collision Resistant hash functions.

5.1. **Definition.** We define a **Hybrid-Collision Resistant Hash Function, (HCR)** H, to be a hash function in which the game of Figure 14 is hard, for fixed values l_1 and l_2. Moreover, we say H is a (T, ϵ)-HCRHF if an adversary with complexity T wins the game on Figure 14 with probability at most ϵ.

Furthermore, we call the pair $(L, M\|K)$ a *hybrid-collision*. Note that, if $l_2 = 0$, then HCR is equivalent to CR. On the other hand, HCR is very close to SPR when $l_1 = 0$. In fact, HCR is interpolating between CR and SPR. This suggests that, finding hybrid-collisions is harder than collisions, but not harder than second-preimages. We will investigate this matter in more detail in the next Section.

Oscar		Challenger
Choose M, $\|M\| = l_1$.	$\xrightarrow{M}$	Choose $K \in_R \{0,1\}^{l_2}$.
	$\xleftarrow{K}$	
Choose L, $\|L\| = l_1 + l_2$.	$\xrightarrow{L}$	Oscar wins if $L \neq M\|K$ and $H(M\|K) = H(L)$.

FIGURE 14. HCR Game

5.2. **On the Difficulty of the HCR Game.** As far as we know, the problem of finding hybrid-collisions has not been addressed in the literature, yet. Here, we investigate this problem in the Random Oracle Model. This gives us an intuition about the difficulty of the problem compared to finding collisions or second-preimages.

Let H be a hash function randomly chosen from $\mathcal{F}^{\mathcal{X},\mathcal{Y}}$, where $\mathcal{X} = \{0,1\}^{l_1+l_2}$ is the set of all possible binary strings of size $l_1 + l_2$ and $|\mathcal{Y}| = 2^k$. Assume that, we are only permitted oracle access to H, i.e., the only way to compute $H(x)$ is to query the value x to the oracle. Further, assume that the adversary, Oscar, is able to access the Random Oracle T times, where $T = 2^t$.

In order to analyze the difficulty of the HCR Game, we find an upper bound on the probability ϵ of Oscar wining the HCR Game.

Let distinct random values $X_1, X_2, \ldots, X_T$ be Oscar's inputs to the random oracle. Moreover, let the hybrid-collision be $(L, M\|K)$. We write $X_i = M_i\|K_i$, where $|K_i| = l_2$ and $|M_i| = l_1$, for all $i = 1, \ldots, T$.

When Oscar wins, there are two cases to consider:

Case 1. $M\|K$ is a random value that happens to collide with $L = X_j$, for some j, $1 \leq j \leq T$.

Case 2. $M\|K$ is a precomputed value, X_i, that collides with $L = X_j$, for some i and j, $1 \leq i, j \leq T$ and $X_i \neq X_j$.

Let us denote the probability of Case 1 and 2 happening by ϵ_1 and ϵ_2, respectively.

In the first case, the probability that $H(M\|K) = X_j$ for each j is 2^{-k}. Hence, the probability of occurrence of one collision is $\epsilon_1 = 1 - (1 - 2^{-k})^T$. If $T = 2^t$ is small compared to 2^k, then ϵ_1 is approximately 2^{t-k}.

The analysis in the second case is more complicated. We find an upper bound on ϵ_1 by estimating the probability ϵ_1' that, for randomly chosen X_i and X_j, $H(X_i) = H(X_j)$. We find this upper-bound by means of properties of a graph G that we later

define. Moreover, we write $X_i = M\|K$, where $|K| = l_2$ and $|M| = l_1$. Note that, randomly choosing X_i is equivalent to randomly choosing M and K separately.

Construct a graph G with $V(G)$ and $E(G)$, denoting the set of vertices and edges respectively, where $V(G) = \{X_1, X_2, \ldots, X_T\}$. Moreover, for any m and n, $m \neq n$, $X_m X_n \in E(G)$ if and only if $H(X_m) = H(X_n)$.

Note that, the maximum number of edges of G is of order $T^2/2$. Furthermore, for any randomly chosen X_m and X_n, the probability that $X_m X_n$ is an edge is 2^{-k}. Hence, the expected number of edges of G is $2^{-k}T^2/2 = 2^{2t-k-1}$. In addition, the expected number of vertices of positive degree is at most 2^{2t-k}.

For each $M \in \{0,1\}^{l_1}$, define f_M as follows

$$f_M = \{m : 1 \leq m \leq T, M_m = M \text{ and } deg(X_m) \geq 1\}.$$

By just considering the restriction $deg(X_m) \geq 1$, we obtain that $|f_M| \leq 2^{2t-k}$.

Now, suppose $K \in \{0,1\}^{l_2}$ is chosen randomly. The probability that $K = K_n$, for $1 \leq n \leq T$, and $n \in f_M$ is less than or equal to 2^{2t-k-l_2}. That is, 2^{2t-k-l_2} is an upper bound on the probability that the $M\|K$ is among the queried values and $H(M\|K)$ is equal to $H(X_j)$ for some j. Hence, $\epsilon_1 \leq \epsilon_1' \leq 2^{2t-k-l_2}$.

Hence, provided that 2^t is small compared to 2^k, we conclude

$$\epsilon = \epsilon_1 + \epsilon_2 \leq 2^{t-k} + 2^{2t-k-l_2}.$$

In Section 5.4 we examine p, the overall success probability of the adversary, given particular values for parameters k, t and l_2.

5.3. **A new Non-Interactive Message Authentication Protocol based on HCR hash functions.** Let H be a HCR hash function and consider the following proposed NIMAP.

1. On input (M, Bob), Alice chooses $K \in_R \{0,1\}^k$ uniformly at random.
2. Alice sends (M, K) to Bob over the broadband channel.
3. Bob receives (M', K').
4. Alice computes $h = H(M\|K)$ and sends h to Bob over the authenticated channel.
5. Bob receives h'.
6. Bob outputs (Alice, M') if $h' = H(M'\|K')$, and rejects otherwise.

The above NIMAP is also depicted in Figure 15.

In this NIMAP, $m_1 = H(M\|K) = h$ and $m_2 = (M, K)$ for a random key K. Moreover, for any M', K' and h', $reconstruct(h', (M', K')) = M'$ if $h' = H(M'\|K')$, and $reconstruct(h', (M', K')) = \perp$ otherwise.

Alice		Bob
Input $(M$, Bob$)$, $\|M\| = l_1$,		
Choose $K \in_R \{0,1\}^{l_2}$.	$\xrightarrow{M, K}$	Receive M', K'.
Compute $h = H(M\|K)$.	$\xRightarrow{h}$	Receive h', accept if $h' = H(M'\|K')$, reject otherwise.

FIGURE 15. The New NIMAP

Clearly, this $(split, reconstruct)$ satisfies the Property (i) of Section 2. That is, any message M can be uniquely recovered:

$$reconstruct(split(M)) = reconstruct((M, K), H(M\|K)) = M.$$

Next we need to show that our $(split, reconstruct)$ satisfies the Property (ii) of Section 2 which says: It is computationally infeasible to find a message M such that given (m_1, m_2), where $split(M) = (m_1, m_2)$, one can efficiently find an $m_2' \in \mathcal{M}_2 \setminus \{m_2\}$ so that $reconstruct(m_1, m_2') \in \mathcal{M}$ with non-negligible probability.

We substitute for the *split* and *reconstruct* functions and restate the Binding Property for our NIMAP as follows:

It is computationally infeasible to find a message M, $|M| = l_1$, such that given $H(M\|K)$ and K, $K \in_R \{0,1\}^{l_2}$, one can efficiently find an L of size $l_1 + l_2$, $L \neq M\|K$, so that $H(L) = H(M\|K)$ with non-negligible probability.

This is exactly what it means for the HCR Game to be hard. Assuming that H is a (T, ϵ)-HCRHF, we conclude that $(split, reconstruct)$ of this NIMAP is (T, ϵ)-binding. Hence, we get the following Corollary of Theorem 1.

Corollary 1. *Let H be a (T, ϵ)-HCRHF. Any adversary against the NIMAP of Figure 15, with online complexity q and offline complexity T, has a probability of success p at most $q\epsilon$.*

5.4. **Parameter sizes.** Let $T = 2^t$ and q be the offline and online complexities respectively. That is, the adversary is allowed to use T hash computations and make Alice send q messages to Bob. Moreover, H be a (T, ϵ)-HCRHF and k be the size of H.

According to Corollary 1, an adversary attacking our proposed NIMAP, using T hash computations and q messages, has probability of success $p \leq q\epsilon$.

In [PV06], Pasini and Vaudenay assume that $q \leq 2^{10}$ and $t \leq 70$. They also require the probability of success of the adversary against the protocol of Figure 13

be less that 2^{-20}. For this to happen, one needs to authenticate 100 bits. That is $k = 100$.

Using the same parameters, $q \leq 2^{10}, t \leq 70$, and $k = 100$ we obtain that $\epsilon \approx 2^{-30} + 2^{40-l_2}$. In order to achieve the same level of security obtained in [PV06], i.e. $p \leq 2^{-20}$, we should have $\epsilon \approx 2^{-30}$. Thus, if we let $l_2 \geq 100$ in our protocol of Figure 15, then we obtain the same level of security of the protocol of Figure 13. That is, the amount of information sent over the authenticated channel is the same as in the Pasini-Vaudenay protocol.

We can actually reduce the size of l_2 to 70 in expense of authenticating one more bit. That is, $q \leq 2^{10}, t \leq 70, k = 101$, and $l_2 = 70$ achieves the same level of security $p \leq 2^{-20}$.

Although we are quite flexible about the amount of information sent over the broadband channel, one should still look into it, at least, as a secondary factor. In our protocol, $l_1 + l_2$ bits are being sent over the insecure channel, where l_1 is the size of the message. However, protocol of Figure 13, requires sending $O(N^2)$ bits over the insecure channel, where N is the size of the message. Hence, our proposed NIMAP requires a lot less bits to be sent over the insecure channel.

5.5. **Advantages of the proposed NIMAP.** Our proposed NIMAP of Figure 15 benefits from a simple and easy to implement structure. It is based on a single assumption that HCR hash functions exist. We do not use any commitment scheme or require any public parameters available to users such as the CRS.

The amount of information sent over the authenticated channel is as low as the most secure NIMAP proposed so far, while achieving the same level of security.

In addition, the amount of information sent over the insecure channel is reduced significantly.

6. Conclusion

We assumed that there are two channels available for communication, one insecure broadband channel and one authenticated narrow-band channel. We produced the required formalism needed in a general model of non-interactive Message Authentication Protocols using these two channels. GNIMAP depicts a general non-interactive Message Authentication Protocol. We proved that GNIMAP is secure given that a Binding Game is hard to win for an adversary with certain properties. Theorem 1 summarizes the security result about GNIMAP.

Further, we examined the NIMAPs found in the literature. We discussed their security in our general model.

Last but not least, we proposed a particular NIMAP based on HCR hash functions. We proved that our proposed NIMAP is secure in the general model given that the HCR Game is hard to win.

Our proposed NIMAP, sends the same amount of information over the authenticated channel as the most secure NIMAP proposed so far, while achieving the same level of security. In comparison with this latter protocol, our NIMAP reduces the amount of information sent over the insecure channel significantly.

Acknowledgements

Douglas R. Stinson's research is supported by NSERC discovery grant 203114-06. Atefeh Mashatan is supported by an NSERC PGSD Scholarship.

References

[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.

[BSSW02] Dirk Balfanz, Diana K. Smetters, Paul Stewart, and H. Chi Wong. Talking to strangers: Authentication in ad-hoc wireless networks. In *Network and Distributed Sytem Security Symposium*, 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 2001: Proceedings of the 8th ACM conference on Computer and Communications Security*, pages 206–214, Philadelphia, Pennsylvania, USA, 2001. ACM Press.

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

[GN04] C. Gehrmann and K. Nyberg. Security in personal area networks. *Security for Mobility, IEE, London*, pages 191–230, 2004.

[NSS] M. Naor, G. Segev, and A. Smith. Tight bounds for unconditional authentication protocols in the manual channel and shared key models. Cryptology eprint archive, report 2006/175, http://eprint.iacr.org/2006/175.pdf.

[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.

[PV06] Sylvain Pasini and Serge Vaudenay. An optimal non-interactive message authentication protocol. In David Pointcheval, editor, *Topics in Cryptography*, volume 3860 of *Lecture Notes in Computer Science*, pages 280–294, San Jose, California, U.S.A., February 2006. Springer-Verlag.

[Vau05] Serge Vaudenay. Secure communications over insecure channels based on short authenticated strings. In Victor Shoup, editor, *Advances in Cryptography*, volume 3621 of *Lecture Notes in Computer Science*, pages 309–326, Sanra Barbara, California, U.S.A., August 2005. Springer-Verlag.