Approximate Message Authentication and Biometric Entity Authentication*

G. Di Crescenzo[1], R. Graveman[2], R. Ge[3], and G. Arce[3]

[1] Telcordia Technologies, Piscataway, NJ
giovanni@research.telcordia.com
[2] Work done while at Telcordia Technologies
rfg@acm.org
[3] University of Delaware, Newark, DE
{ge, arce}@ece.udel.edu

Abstract. Approximate Message Authentication Code (AMAC) is a recently introduced cryptographic primitive with several applications in the areas of cryptography and coding theory. Briefly speaking, AMACs represent a way to provide data authentication that is tolerant to acceptable modifications of the original message. Although constructs had been proposed for this primitive, no security analysis or even modeling had been done.

In this paper we propose a rigorous model for the design and security analysis of AMACs. We then present two AMAC constructions with desirable efficiency and security properties.

AMAC is a useful primitive with several applications of different nature. A major one, that we study in this paper, is that of entity authentication via biometric techniques or passwords over noisy channels. We present a formal model for the design and analysis of biometric entity authentication schemes and show simple and natural constructions of such schemes starting from any AMAC.

1 Introduction

The rise of financial crimes such as identity theft (recent surveys show there are currently 7-10 million victims per year) and check fraud (more than 500 million checks are forged annually with losses totaling more than 10 Billion dollars in the United States alone) is challenging financial institutions to meeting high security levels of entity authentication and data integrity. Passwords are a good start to secure access to their systems but, when used alone, don't seem

* Copyright Telcordia Technologies. Prepared through collaborative participation in the Communications and Networks Consortium sponsored by the U. S. Army Research Laboratory under the Collaborative Technology Alliance Program, Cooperative Agreement DAAD19-01-2-0011. The U. S. Government is authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation thereon.

A.S. Patrick and M. Yung (Eds.): FC 2005, LNCS 3570, pp. 240–254, 2005.

enough to provide the security and convenience level for identification needed by financial organizations. (Passwords can be compromised, stolen, shared, or just forgotten.) Biometrics, on the other hand, are based on a user's unique biological characteristics, and can be an effective *additional* solution to the entity authentication problem for financial systems. One challenge in implementing biometric authentication is, however, the reliability of the system with respect to errors in repeated measurements of the same biometric data, such as fingerprints, voice messages, or iris scans.

In this paper we put forward a formal model for the study of approximate data authentication schemes, that are tolerant with respect to errors in the data, and therefore are suitable for the verification of biometric data in entity authentication schemes. We then present efficient constructions of approximate data authentication, and use them to obtain efficient constructions for two types of biometric entity authentication schemes.

Data Authentication. A fundamental cryptographic primitive is that of Message Authentication Codes (MAC), namely, methods for convincing a recipient of a message that the received data is the same that originated from the sender. MACs are extremely important in today's design of secure systems since they reveal to be useful both as atomic components of more complex cryptographic systems and as themselves alone, to guarantee integrity of stored and transmitted data. Traditional message authentication schemes create a hard authenticator, where modifying a single message bit would result in a modification of about half the authentication tag. These MACs fit those applications where the security requirement asks to reject any message that has been altered to the minimal extent. In many other applications, such as those concerning biometric data, there may be certain modifications to the message that may be acceptable to sender and receiver, such as errors in reading biometric data or in communicating passwords through very noisy channels. This new scenario, not captured by the traditional notion of MACs, motivated the introduction and study in [6] of a new cryptographic primitive, a variant of MACs, which was called Approximate Message Authentication Code (AMAC); namely, methods that propagate "acceptable" modifications to the message to "recognizable" modifications in the authentication tag, and still retain their security against other, "unacceptable" modifications. Examples of the applicability of AMACs include: message authentication in highly-noisy or highly-adversarial communication channels, as in mobile ad hoc networks; simultaneous authentication of sets of semantically equivalent messages; and, of specific interest in this paper, entity authentication through inherently noisy data, such as biometrics or passwords over noisy channels.

Our Contributions. If, on one hand, after investigations in [6, 17], the intended notion of AMAC was precisely formulated, on the other hand, a rigorous model for the security study of AMACs was not. Therefore, a problem implicitly left open by [6, 17] was that of establishing such a model. In this paper we propose a rigorous model for analyzing approximation in message authentication. It turns out that the issue of approximation has to be considered in both the correctness property (if Alice and Bob share a key and follow the protocol, then Bob accepts the message) and the security property (no efficient adversary not knowing the shared key and mounting a chosen message attack can make Bob accept a new message). Our notions of approximate correctness and approximate security use as a starting point the previously proposed notions for conventional MACs and address one difficulty encountered in both allowing acceptable modifications to the message and achieving a meaningful security notion. In addition, we formulate two preimage-resistance requirements that make these AMACs especially applicable to two variants of biometric entity authentication problems.

We then present two AMAC constructions: the first scheme uses systematic error correcting codes, is stateless and satisfies our weaker notion of preimage resistance; the second scheme solves the technical problem of constructing a probabilistic universal one-way hash function with distance-preserving properties, is counter-based and satisfies our stronger notion of preimage resistance. Both constructions can be implemented quite easily and only use symmetric primitives.

We then show how to apply these constructions (and, in fact, any AMAC scheme) to obtain simple and efficient biometric entity authentication schemes in both a closed-network and an open-network setting, for which we also present a formal model. Our scheme are non-interactive and can be seen as an extension, using biometrics, of well-known password-based entity authentication schemes.

Formal proofs and some definitions are only briefly sketched due to lack of space.

Related Work. References in conventional Message Authentication Codes are discussed in Section 2. Universal one-way hash function were introduced in [14] and are being often applied in cryptographic constructions. Related work to AMACs includes work from a few different research literatures.

There is a large literature that investigates biometric techniques without addressing security properties (see, e.g. [8] and references therein). Security and privacy issues in biometrics have been independently recognized and advocated by many researchers (see, e.g., [3, 15, 16]).

A second literature (related to information and coding theory) investigates techniques for authentication of noisy multimedia messages (see, e.g., [12, 13] and references therein). All these constructs either ignore security issues or treat them according to information theoretic models. Typically, constructions of the latter type have a natural adaptation to the symmetric MAC setting but all constructions we found, after this adaptation, fail to satisfy the MAC requirement of security under chosen message attack (and therefore the analogue AMAC requirement). Some works use digital signatures as atomic components but they result in constructions that are not preimage-resistant, according to our Definition 2, and therefore cannot be applied to give a satisfactory solution to our biometric authentication problem.

A third literature investigates coding and combinatorial techniques for error tolerance in biometrics (see, e.g., [10, 9]), as well as privacy amplification from reconciliation. Recently, [5, 2] considered the problem of generating strongly random keys from biometric data. Although these constructions might be useful towards solving the problem of biometric entity authentication, current proposals fall short of achieving this. In particular, the proposal in [5] was broken by [2] in the setting of identification to multiple servers; and the (interactive) proposal of [2] is still based on some (somewhat questionable) assumption referring to biometrics as entropy sources. Yet, these papers address interesting primitives and notions (fuzzy commitments, fuzzy extractors, etc.) unaddressed by ours and viceversa. Our non-interactive proposal is based on a very intuitive and perhaps minimal assumption on biometrics.

We stress that all this previous work did not even imply a formal definition of AMACs.

2 Definitions and Preliminaries

In this section we present our novel definition of Approximate MACs. In the rest of the paper we will assume familiarity with definitions of cryptographic primitives used in the paper, such as universal one-way hash functions, (conventional) MACs, symmetric encryption schemes and finite pseudo-random functions.

Approximation in MACs. We introduce formal definitions for approximate MACs, using as a starting point the well-known definition for conventional MACs. Informally, one would like an approximate MAC to be tolerant to "acceptable" modifications to the original message. Less informally, we will define approximate versions of the same properties as an ordinary MAC, where the approximation is measured according to some polynomial-time computable distance function on the message space. For the correctness property, the notion of a modification being acceptable is formalized by requiring an authentication tag computed for some message m, to be verified as correct even for messages having up to a given distance from m. We note that this property might not be compatible with the property of security against chosen message attack, for the following reason. The latter property makes an adversary unable to produce a valid pair of message and authentication tag, for a new message, for which he hasn't seen an authentication tag so far; the former property, instead, requires the receiver himself to be able to do so for some messages, that is, for messages having a certain distance from the original message obtained from the sender. In order to avoid this apparent definitional contradiction, we define a chosen message attack to be successful if the valid pair of message and authentication tag produced by the adversary contains a message which has a larger distance from all messages for which he has seen an authentication tag during his chosen message attack. Therefore, we even define the security property for MACs in some approximate sense. We now proceed more formally.

Definition 1. Let M denote the message space and let d be a polynomial-time computable distance function over M. An *approximately correct and approximately secure message authentication code for distance function d* (briefly, d-ac-as-MAC) is a triple (Kg,Tag,Verify), where the polynomial-time algorithms

Kg, Tag, Verify satisfy the following syntax. The key-generation algorithm Kg takes as input a security parameter 1^l, and distance function d, and returns an l-bit secret key k. The authenticating algorithm Tag takes as input a message m, a secret key k, and distance function d, and returns a string *tag*. The verifying algorithm Verify takes as input a message m, a secret key k, a string *tag*, and distance function d, and returns a value $\in$ {yes,no}. Moreover, the triple (Kg,Tag,Verify) satisfies the following two requirements.

1. (d, p, δ)-*Approximate Correctness*: after k is generated using Kg, if *tag* is generated using algorithm Tag on input message m and key k, then, with probability at least p, algorithm Verify, on input k, m', tag, outputs: *yes*, if $d(m, m') \leq \delta$.
2. $(d, \gamma, t, q, \epsilon)$-*Approximate Security*: Let k be generated using Kg; for any algorithm *Adv* running in time at most t, if *Adv* queries algorithm Tag$(k, \cdot)$ with adaptively chosen messages, thus obtaining pairs $(m_1, t_1), \ldots, (m_q, t_q)$, and then returns a pair (m, t), the probability that Verify$(k, m, t) =$ *yes* and $d(m, m_i) \geq \gamma$ for $i = 1, \ldots, q$, is at most ϵ.

Note that (t, q, ϵ)-secure MAC schemes are (d, p, δ)-approximately correct and $(d, \gamma, t, q, \epsilon)$-approximately secure MAC schemes for $p = 1$, $\delta = 0$, $\gamma = 1$, and d equal to the Hamming distance. In the sequel, we will omit d in the term d-ac-as-MAC when clear from the context, or directly abbreviate the term d-ac-as-MAC as AMAC. Although not included in the above definition, as for conventional MACs, an important *efficiency requirement* for AMACs is that the size of the tag is desired to be significantly smaller than the length of the input message.

Two Additional Properties of AMACs. In certain applications of AMACs as those considered in this paper, it may be desirable that the AMAC tag does not help in recovering any message for which that tag is valid. We formally define two variants of a 'preimage-resistance' property. In the first variant, called 'weak preimage-resistance', we require that the tagging algorithm, if viewed as a function on the message space, is hard to invert, no matter what is the distribution on the message space. (Later, while showing the applications of AMACs to biometric entity authentication, this property will be useful in proving that the entity authentication scheme obtained is secure against adversaries that can gain access to the AMAC output from the biometric storage file.) In the second variant, called 'strong preimage-resistance', we require that this property holds even if the adversary is given access to the receiver's private key. We now formally define both properties.

Definition 2. The d-ac-as-MAC (Kg,Tag,Verify) is (d, t, q, ϵ)-*weakly-preimage-resistant* if the following holds. Let k be generated using Kg; and assume that an efficient algorithm *Adv* obtains from an oracle $\mathrm{O}(d, k)$ valid tags $t_1, \ldots, t_q$; that is, tags for which there exist messages $m_1, \ldots, m_q$, independently drawn from some efficiently samplable distribution D_m, such that $t_i =$Tag(d, k, m_i), for $i = 1, \ldots, q$. For any such *Adv* running in time at most t, the probability that $Adv(d, M, t_1, \ldots, t_q)$ returns m' such that Verify$(d, k, m', t_i) = 1$ for some $i \in \{1, \ldots, q\}$, is at most ϵ. Furthermore, we say that the d-ac-as-MAC (Kg,Tag,Verify) is (t, ϵ)-*strongly-preimage-resistant* if the above holds even with respect to algorithms *Adv* who takes k as an additional input.

We note that essentially all conventional MAC constructions in the literature would satisfy an analogue preimage-resistance requirement. However it is easy to transform a MAC into one that is not weakly preimage-resistant and for some applications like biometric identification, it may be very desirable to require that the AMAC used is weakly or strongly preimage-resistant (or otherwise an accidental loss of the AMAC output or the server's private key could reveal a password or some biometric data to an adversary).

Previous Work on AMACs. Previously to this work, variations of a single approximate MAC contruction had been proposed and investigated in [6, 17]. Informally, the tagging algorithm in these constructions uses operations such as xoring the message with a pseudo-random string of the same length, computing a pseudo-random permutation of the message, and returning majority values of subsets of message bits. As already observed in [4], it can be seen that these constructions are secure against an adversary that cannot mount a chosen message attack; while they are not intended to be secure under a sufficiently long chosen message attack, since they only use a polynomial amount of pseudo-randomness.

Simple Attempts Towards AMAC Constructions. First of all, we remark that several simple constructions using arbitrary error correcting codes and ordinary MACs fail in satisfying even the approximate correctness and security requirements of AMACs. These include techniques such as interpreting the input message as a codeword, and using a conventional MAC to authenticate its decoding (here, the property of approximate correctness fails). Other techniques that also fail are similar uses of fuzzy commitments from [10], fuzzy sketches from [5] and reusable fuzzy extractors from [2]. We note however that there are a few simple constructions that meet the approximate correctness and security requirements of AMACs but don't meet the preimage-resistance or the efficiency requirements. The simplest we found goes as follows. Let us denote as (K,T,V) a conventional MAC scheme. The tagging algorithm, on input key k and message m, returns $tag = m \,|\, \mathrm{T}(k, m)$. The verifying algorithm, on input k, m', tag, sets $tag = t1 \,|\, t2$ and returns 1 if and only if $d(t1, m') \leq \delta$ and $V(k, t1, t2) = 1$, where d is the distance function. The scheme satisfies the approximate correctness and security; however, note that the tag of this scheme contains the message itself and therefore the scheme is neither preimage-resistant nor efficient.

3 Our AMAC Constructions

In this section we present two constructions of approximately-correct and approximately secure MACs with respect to the Hamming distance. The first construction is stateless and weakly preimage-resistant under the existence of secure symmetric encryption schemes and weakly preimage-resistant conventional

MACs. The second construction, the main one in the paper, is counter-based and strongly preimage-resistant under the existence of collision-intractable hash functions.

3.1 A Weakly Preimage-Resistant AMAC Construction

A construction of an AMAC for the Hamming distance function can be obtained by using any conventional MAC scheme, any symmetric encryption scheme, and any appropriate systematic error correcting code. The construction satisfies approximate correctness with optimal parameter $p = 1$ and approximate security with optimal parameter $\gamma = \delta + 1$.

Formal Description. Let us denote by (K$_a$,T,V) a conventional MAC scheme, and by (K$_e$,E,D) a symmetric encryption scheme. Also, by (SEnc,SDec) we denote a systematic error-correcting code (that is, on input m, SEnc$(m) = c$, where $c = m|pc$, and pc are parity check bits), such that the decoding algorithm perfectly recovers the message if at most δ errors happened or returns failure symbol $\perp$ otherwise (this latter condition is without loss of generality as any error correcting code can be simply transformed into one that satisfies it).

Instructions for Kg: generate a uniformly distributed k-bit key K

Input to Tag: two k-bit keys K_a, K_e, an n-bit message M, parameters p, δ, γ.

Instructions for Tag:

```
1. Set $c = Enc(M)$ and write $c$ as $c = M|pc$
2. Set $subtag = T_{K_a}(M)$ and $epc = E(K_e, pc)$
3. **Return:** $tag = epc|subtag$ and halt.
```

Input to Verify: parameters p, δ, γ, two k-bit keys K_a, K_e, an n-bit message M' and a string tag

Instructions for Verify:

```
1. Write $tag$ as $tag = epc|subtag$
2. Let $pc = D(K_e, epc)$ and $m' = Dec(M'|pc)$
3. If $m' = \perp$ then **Return:** 0
4. If $V(K_a, m', subtag) = 1$ then **Return:** 1 else **Return:** 0.
```

We can prove the following

Theorem 1. Let d denote the Hamming distance, let n be the length of the input message for (Kg,Tag,Verify) and let (SEnc,SDec) a systematic error-correcting code that corrects up to δ errors and returns $\perp$ if more than δ errors happened, for some parameter δ. Then (Kg,Tag,Verify) is an AMAC that satisfies the following properties:

1. (d, p, δ)-approximate correctness for $p = 1$
2. $(d, \gamma, t', q', \epsilon')$-approximate security under the assumption that (KG$_a$,T,V) is a (t, q, ϵ)-secure MAC, where $\gamma = \delta + 1$, $t' = t - O(q \cdot time(\mathrm{D}) + time(\mathrm{SDec}))$, $q' = q$, $\epsilon' = \epsilon$, and $time(F)$ denotes the running time of function F.

3. (d, t', q', ϵ')-weak preimage-resistance under the assumption that (KG$_a$,T,V) is (t_a, q_a, ϵ_a)-weakly preimage-resistant and (KG$_e$,E,D) is a (t_e, q_e, ϵ_e)-secure symmetric encryption scheme (in the real-or-random sense), where $q_e = 1$, $q' = q_a$, $\epsilon' \leq \epsilon_a + q_e\epsilon_e$, and $t' = \min(t_1, t_2)$, for $t_1 = t_a - O(q' \cdot (\mathit{time}(\mathrm{Enc}) + \mathit{time}(\mathrm{E}) + \mathit{time}(D_m)) + \mathit{time}(\mathrm{KG}_e))$, and $t_2 = t_e - O(q' \cdot (\mathit{time}(\mathrm{Enc}) + \mathit{time}(\mathrm{T}) + \mathit{time}(D_m)) + \mathit{time}(\mathrm{KG}_a))$.

The above theorem already provides AMACs with some useful properties, such as approximate correctness, approximate security and weak preimage-resistance. However, we note two facts that make this scheme not a definitely satisfactory solution: first, its tag length depends on the performance of the systematic code used, and can thus be significantly longer than regular MACs even for moderately large values of the parameter δ; second, this scheme does not satisfy the stronger preimage resistance property. As we will see in Section 4, the latter is very desirable in order to construct a network biometric entity authentication scheme, a main application of AMACs in this paper. The scheme in Section 3.2 satisfies both efficiency of tag length (for any value of δ) and the strong preimage-resistance property.

3.2 Our Main AMAC Construction

Informal Description. We explain the ideas behind this scheme in two steps. First, we explain how to construct a probabilistic universal one-way hash function and use it to guarantee that outputs from this hash function will have some additional distance-preserving properties. Second, we construct an approximately correct and secure MAC based on such a probabilistic universal one-way hash function.

We achieve a combination of distance-preserving properties and target collision resistance by making a universal one-way hash function probabilistic, and using the following technique. First, the message bits are xored with a pseudo-random string and pseudo-randomly permuted and then the resulting message is written as the concatenation of several equal-size blocks. Here, the size of each block could be the fixed constant size (e.g., 512 bits) of the input to compression functions (e.g., SHA) that are used as atomic components of practical constructions of universal one-way hash functions. Now multiple hashes are computed, each being obtained using the universal one-way hash function, using as input the concatenation of a different and small enough subset of the input blocks. Here, the choice of each subset is done using pseudo-random bits. Furthermore, each subset has the same size, depending on the length of the input and on the desired distance-preserving properties. The basic idea so far is that by changing the content of some blocks of the message, we only change a small fraction of the inputs of the atomic hashes and therefore only a small fraction of the outputs of those hashes will change.

Given this ‘probabilistic universal one-way hash function’, the tagging and verifying algorithm can be described as follows.

The tagging algorithm, on input a random key and a message, uses another value, which can be implemented as a counter incremented after each application (or a random value chosen independently at each application). Then the algorithm computes the output of the finite pseudo-random function on input such value and divides this output in two parts: the first part is a random key for the universal one-way hash function and the second part is a sequence of pseudo-random bits that can be used as randomness for the above described probabilistic universal one-way hash function. Now, the tagging algorithm can run the latter function to compute multiple hashes of the message. The tag returned is then the input to the finite pseudo-random function and the hashes.

The construction of the verifying algorithm is necessarily differently from the usual approach for exactly correct and secure MACs (where the verifying algorithm runs the tagging algorithm on input the received message and checks that its output is equal to the received tag), as this algorithm needs to accept the same tag for multiple messages. Specifically, on input the tag returned by the tagging algorithm, the verifying algorithm generates a key and pseudo-random bits for the probabilistic universal one-way hash function and computes the hashes of the received message exactly as the tagging algorithm does. Finally, the verifying algorithm checks that the received and the computed sequences of hashes only differ in a small enough number of positions.

Formal Description. Let k be a security parameter, t be an approximation parameter, and c be a block size constant. We denote by $H = \{tcrh_K : K \in \{0,1\}^k\}$ a finite universal one-way hash function (also called 'target collision resistance function' in the literature), such that for each $K \in \{0,1\}^k$, $tcrh_K$ is a collision-intractable hash function. We denote by $F = \{f_K : K \in \{0,1\}^k\}$ a finite pseudo-random function. We now present our construction of an approximately-secure and approximately-correct MAC, which we denote as (Kg,Tag,Verify).

Instructions for Kg: generate a uniformly distributed k-bit key K

Input to Tag: a k-bit key K, an n-bit message M, parameters p, δ, γ, a block size 1^c and a counter ct.

Instructions for Tag:

```
– Set $x_1 = \lceil n/2c\delta \rceil$ and $x_2 = \lceil 10\log(1/(1-p)) \rceil$
– Set $(u|\pi|\rho|L) = f_K(ct)$, where $u \in \{0,1\}^k$, $L \in \{0,1\}^n$, and $\pi$ is a permutation of $\{0,1\}^n$
– Write $\pi(L \oplus M)$ as $M_1|\cdots|M_{\lceil n/c \rceil}$, where $|M_i| = c$ for $i = 1, \ldots, \lceil n/c \rceil$
– Use $\rho$ as randomness to randomly choose $x_1$-size subsets $S_1, \ldots, S_{x_2}$ of $\{1, \ldots, \lceil n/c \rceil\}$
– For $i = 1, \ldots, x_2$,
      let $N_i = M_{i_1}|\cdots|M_{i_{x_1}}$, where $S_i = \{i_1, \ldots, i_{x_1}\}$
      let $sh_i = tcrh_u(N_i)$
– Let $subtag = sh_1|\cdots|sh_{x_2}$
– Return: $tag = ct|subtag$.
– Set $ct = ct + 1$ and halt.
```

Input to Verify: parameters δ, γ, a block size 1^c, a k-bit key K, an n-bit message M' and a string tag

Instructions for Verify:

```
– Write $tag$ as $ct|sh_1|\cdots|sh_{x_2}$
– Set $x_1 = \lceil n/2c\delta \rceil$ and $x_2 = \lceil 10\log(1/(1-p)) \rceil$
– Set $(u|\pi|\rho|L) = f_K(ct)$, where $u \in \{0,1\}^k$, $L \in \{0,1\}^n$, and $\pi$ is a permutation of $\{0,1\}^n$
– Write $\pi(L \oplus M')$ as $M'_1|\cdots|M'_{\lceil n/c \rceil}$, where $|M'_i| = c$ for $i = 1, \ldots, \lceil n/c \rceil$
– Use $\rho$ to randomly select $x_1$-size subsets $S'_1, \ldots, S'_{x_2}$ of $\{1, \ldots, \lceil n/c \rceil\}$
– For $i = 1, \ldots, x_2$,
    let $N'_i = M'_{i_1}|\cdots|M'_{i_{x_1}}$, where $S'_i = \{i_1, \ldots, i_{x_1}\}$
    let $sh'_i = tcrh_u(N'_i)$
– Check that $sh'_i = sh_i$, for at least $\alpha x_2$ of the values of $i \in \{1, \ldots, x_2\}$, for $\alpha = 1 - 1/2\sqrt{e} - 1/2e$.
– Return: 1 if all verifications were successful and 0 otherwise.
```

The above construction satisfies the following

Theorem 2. *Let d denote the Hamming distance, let δ, c, p be parameters. Then the above construction (Kg,Tag,Verify) is an AMAC satisfying the following properties.*

1. (d, p, δ)-approximate correctness
2. $(d, \gamma, t_A, q_A, \epsilon_A)$-approximate security under the assumption that F is a (t_F, q_F, ϵ_F)-secure pseudo-random function and H is a (t_H, q_H, ϵ_H)-target-collision-resistant hash function, where $\gamma = 2\delta$, $\epsilon_A \leq p_1 \leq \epsilon_F + 2\epsilon_H \cdot q_A + 2(1-p)$, $q_A = q_F \geq 1$, $q_H = \lceil 10\log(1/(1-p)) \rceil$, and $t_A = \min(t_{A,1}, t_{A,2})$, where n is the length of the message, c is a block size constant, ct is the counter input to algorithm Tag, $time(g; x)$ denotes the time required to compute function g on inputs of size x, and
   - $t_{A,1} = t_F - O(q_A(n(\log n + \log(1/(1-p))) + \log(1/(1-p)) + time(h_u; n/2c\delta))$
   - $t_{A,2} = t_H - O(n(\log n + \log(1/(1-p))) + time(f_K; |ct|))$.
3. (d, t', q', ϵ')-strong preimage resistance under the assumption that for each $K \in \{0,1\}^k$, function h_K is (t, ϵ)-collision resistant, where $\epsilon' \leq \epsilon$, and $t' = t - O(time(\text{Tag}; n))$.

Remarks. Our scheme is quite simple to implement and our implementation experience required very small effort. We note that in practice the families H and F can be implemented using well-known keyed cryptographic hash functions (e.g., UMAC [1] or other constructions cited in there) and well-known block ciphers (e.g., AES).

The length of the tag returned by algorithm Tag is $x_2 \cdot c$, where $x_2 = 10\log(1/(1-p))$, and c is the length of the output of the universal one-way hash function. (In practice, this value could be smaller, but it would require a more involved security analysis.) We note that c is constant with respect to n, and acceptable settings of parameter p can lie anywhere in the range $[1-1/2^{(\log n)^{1+\epsilon}}, 1]$, for any constant $\epsilon > 0$. Therefore the length of the tag returned by the scheme can be as small as $10c(\log n)^{1+\epsilon}$; most importantly, this holds for *any* value of parameter δ. The tag length remains much shorter than the message even for much larger settings of p; for instance, if $p = 1 - 2^{-\sqrt{n}}$, the tag length becomes $O(\sqrt{n})$.

3.3 Properties of Our Main Construction

We now discuss the properties mentioned in Theorem 2. As the strong preimage resistance property immediately follows from the collision resistance of functions from H, we now focus on proving the approximate correctness and approximate security properties.

Approximate Correctness. Assume $d(M, M') \leq \delta$. First, we assume for simplicity that f_K is a random function. Then, for $i = 1, \ldots, x_2$, define random variable X_i as equal to 1 if $sh_i \neq sh'_i$ or 0 otherwise. Furthermore, we denote by N_i and $M_{i_1}, \ldots, M_{i_{x_1}}$ (resp., N'_i and $M'_{i_1}, \ldots, M'_{i_{x_1}}$) the values used in the 5th step of algorithm Tag on input M (resp., M'). Then it holds that

$$\begin{aligned} a &= \text{Prob}\,[\,X_i = 1\,] \\ &\leq 1 - \text{Prob}\,[\,N_i = N'_i\,] \\ &= 1 - (\text{Prob}\,\left[\,M_{i_1} = M'_{i_1}\,\right])^{n/2\delta} \\ &\leq 1 - ((n/c - \delta)/(n/c))^{n/2c\delta} = 1 - (1 - c\delta/n)^{n/2c\delta} \;\leq\; 1 - 1/\sqrt{e}, \end{aligned}$$

where the first inequality follows from the definition of X_i and from how sh_i, sh'_i are computed; the second equality follows from the definition of N_i, N'_i; and the second inequality follows by observing that M and M' differ in at most δ blocks, and that blocks M_i, M'_i are uniformly and independently chosen among all blocks in $\pi(M), \pi(M')$, respectively, as so are subsets S_i, S'_i. We obtain that $a - \alpha = (\sqrt{e} - 1)/2e$. Since $X_1, \ldots, X_{x_2}$ are independent and identically distributed, we can apply a Chernoff bound and obtain that

$$\text{Prob}\left[\sum_{i=1}^{x_2} X_i \;<\; \alpha x_2\right] \;\leq\; e^{-2(a-\alpha)^2 x_2} \;\leq\; 1 - p,$$

which implies that algorithm Verify returns 1 with probability at least p. Note that the assumption that f_K is a random function can be removed by only subtracting a negligible factor to p, as otherwise the test used by algorithm Verify can be used to contradict the pseudorandomness of F.

Approximate Security. The proof for this (only sketched here) requires the definition of four probability experiments that slightly differ from each other. We assume that the requirement of $(d, \gamma, t, q, \epsilon)$-approximate security is not satisfied and reach some contradiction.

Experiment 1 is precisely the experiment in the definition of approximate security. We denote by p_1 the probability that experiment 1 is successful; our original assumption implies that $p_1 > \epsilon$.

Experiment 2 differs from experiment 1 only in that *Adv* queries a finite random function r rather than a finite pseudo-random function Tag. Denoting as

p_2 the probability that experiment 2 is successful, we can prove that $p_2 - p_1 \leq \epsilon_F$, or otherwise *Adv* can be used to violate the assumption that F is a (t_F, q_F, ϵ_F)-secure pseudo-random function.

Experiment 3 is a particular case of experiment 2; specifically, it is successful when experiment 2 is and the adversary returns a tag with the same counter as in a tag previously returned by the oracle. We distinguish two cases, according to whether the following condition is true or not: all $i \in \{1, \ldots, x_2\}$ such that $sh_i = sh'_i$ are associated with values N_i, N'_i such that $N_i = N'_i$. If the condition does not hold, then this means that *Adv* found two distinct preimages N_i, N'_i of the same output under $tcrh_u$ and therefore *Adv* can be used to violate the assumption that H is a (t_H, q_H, ϵ_H)-target collision resistant hash function. If the condition holds, then this means that a large number of subsets S_i 'missed' all $\gamma = 2\delta$ bits where M and M' differ. By using a Chernoff bound argument dual to that used in the proof of the approximate correctness property, we derive that this happens with probability at most $1 - p$. We denote as p_3 the probability that experiment 3 is successful, and, from the above two cases, obtain that $p_3 \leq \epsilon_H \cdot q_A + 1 - p$.

Experiment 4 is a particular case of experiment 2; but it considers the case complementary to the case in experiment 3. Specifically, it is successful when experiment 2 is and the adversary returns a tag with a counter different from those in all tags previously returned by the oracle. The analysis of this case goes on very similarly as for experiment 3, with the only difference that in the step similar to the proof of the approximate correctness property, we use the fact that the messages M, M' are xored with pseudo-random strings. We obtain that $p_4 < p_3$, where by p_4 we denote the probability that experiment 4 is successful.

We conclude the analysis by using the obtained inequalities: $p_1 - p_2 \leq \epsilon_F$, $p_2 \leq p_3 + p_4$, $p_3 \leq \epsilon_H \cdot q_A + 1 - p$, and $p_4 < p_3$; and therefore obtaining that $\epsilon_A \leq p_1 \leq \epsilon_F + 2\epsilon_H \cdot q_A + 2(1 - p)$.

4 Biometric Entity Authentication

We present a model for the design and analysis of biometric entity authentication (BEA) schemes, and show that two simple constructions based on AMACs can be proved secure in our model under standard assumptions on cryptographic tools and biometric distribution.

Our Model. There is a server S and several users $U_1, \ldots, U_m$, where the server has a biometric storage file bsf and each user U_i is associated with a biometric b_i, a reader R_i and a computing unit CU_i, for $i = 1, \ldots, m$. We define a (non-interactive) BEA scheme between user U_i and S as the following two-phase protocol. The first phase is an *initialization phase* during which user U_i and S agree on various parameters and shared keys and S stores some information on bsf. The second phase is the *authentication phase*, including the following steps. First, user U_i inputs her biometric b_i to the reader R_i, which extracts some feature information $fb_{i,t}$ (this may be a sketched version of the original biometric

b_i) and returns a measurement $mb_{i,t}$, where t here represents the time when R_i is executed. (Specifically, the reader may return a different value $mb_{i,t}$ for each different time t, on input the same b_i.) Then the computing unit CU_i, on input $mb_{i,t}$ sends an authenticating value $ab_{i,t}$ to the server, that, using information stored during the initialization phase, decides whether to accept $ab_{i,t}$ as a valid value for user U_i or not.

The *correctness* requirement for a BEA scheme states that the following happens with high probability: after the initialization phase is executed between $U_i(b_i)$ and S, if, for some t, $mb_{i,t} = R_i(b_i)$, and $ab_{i,t} = CU_i(mb_{i,t})$ then S accepts pair $(U_i, ab_{i,t})$.

An *adversary Adv* tries to attack a BEA scheme by entering a biometric b_j into a reader R_i, and, before doing that, can have access to several and different resources, according to which parties it can corrupt (i.e., noone; users U_j, for $j \neq i$; server S; etc.), and which communication lines or storage data he has access to (i.e., none; the communication lines containing any among $mb_{i,t}, ab_{i,t}$; the biometric storage file bsf; the server's secret keys; user U_i's secret keys, etc.). The *security* requirement for a BEA scheme states that after the initialization phase is executed between $U_i(b_i)$ and S, for $i = 1, \ldots, m$, the probability that an efficient adversary *Adv* can input his biometric b_j into a reader R_i, for $i \neq j$, and make S accept the resulting pair $(U_i, ab^j_{i,t})$, is negligible.

We are now ready to show two simple BEA constructions given any AMAC scheme with certain properties (in fact, not necessarily as strong as those required by Definition 1). The first construction is for *local* BEA; that is, the adversary has no access to the measurements $mb_{i,t}$ and the user can send them in the clear to the server. Local BEA is comparable, in terms of both functionality and security, to well-known password-based authentication schemes in non-open networks. The second construction is for *network* BEA; that is, the message sent from a user to a server during the authentication phase can travel through an open network. Network BEA should be contrasted, in terms of both functionality and security, to password-based authentication schemes in open networks; in particular, we will show that our scheme does not require a user to send over an open network (not even in encrypted form) a reading of her biometric. Both constructions necessarily make an assumption on the distribution of biometric that we now describe.

A Basic Assumptions on Biometrics. We assume that there exist a distance function d, appropriate parameters $\delta < \gamma$, and an efficiently computable measurement M of biometrics such that: (1) for each individual with a biometric b with feature information $fb(t)$ at time t, and for any times t_1, t_2, it holds that $d(\mathrm{M}(fb(t_1)),\mathrm{M}(fb(t_2))) \leq \delta$; (2) for any two individuals with biometrics b_1, b_2, with feature information $fb_1(t), fb_2(t)$ at time t, respectively, and for any times t_1, t_2, it holds that $d(\mathrm{M}(fb(t_1)),\mathrm{M}(fb(t_2))) \geq \gamma$. We refer to this as the *Biometric Distribution Assumption* (BD Assumption). We note that biometric entity authentication (in any model) inherently relies on similar assumptions.

A Construction for Local BEA. Informally, the first construction consists of the user sending the reading of her biometric to the server, that checks it against the previously stored AMAC tag of a reading done at initialization phase. More formally, let (Kg,Tag,Verify) denote an AMAC scheme. Then the BEA scheme lAmacBEA goes as follows. During the initialization phase, user U_i sends ab_{i,t_0} to the server S, that stores tag_0 =Tag(k, ab_{i,t_0}) in the bsf file. During the authentication phase, at time t_1, user U_i inputs b_i into the reader R_i, that returns mb_{i,t_1}; the latter is input to CU_i that returns $ab_{i,t_1} = mb_{i,t_1}$; finally, pair (U_i, ab_{i,t_1}) is sent to S. On input pair (U_i, ab_{i,t_1}), server S computes Verify(k, ab_{i,t_1}, tag_0) and accepts U_i if and only if it is equal to 1.

We can prove the following

Theorem 3. Under the BD assumption, if (Kg,Tag,Verify) is an AMAC scheme then the construction lAmacBEA is a BEA scheme satisfying the above correctness and security requirement against efficient adversaries that can corrupt up to all users U_j but one. Furthermore, if scheme (Kg,Tag,Verify) is weakly preimage-resistant then the construction lAmacBEA satisfies security against efficient adversaries that have also access to the biometric storage file bsf.

A Construction for Network BEA. Informally, the second construction modifies the first construction by having the user compute the AMAC tag over the reading of her biometric; the AMAC tag is then sent to the server that can check it against the previously stored AMAC tag of a reading done at initialization phase. Also, we assume for simplicity that the channel between each user and the server is properly secured (using standard encryption, authentication and time-stamping techniques). More formally, let (Kg,Tag,Verify) denote an AMAC scheme with strong preimage resistance. Then the BEA scheme nAmacBEA goes as follows. During the initialization phase, user U_i inputs her biometric b_i into reader R_i, that returns mb_{i,t_0}; the latter is input to CU_i that returns and sends ab_{i,t_0} =AMAC(k, mb_{i,t_0}) to S; finally, S stores ab_{i,t_0} into bsf. The authentication phase is very similar to the identification phase; specifically, user U_i computes ab_{i,t_1} in the same way, and pair (U_i, ab_{i,t_1}) is sent to S, that computes Verify$(k, ab_{i,t_1}, ab_{i,t_0})$ and accepts U_i if and only if it is equal to 1.

We can prove the following

Theorem 4. Under the BD assumption, if (Kg,Tag,Verify) is an AMAC scheme, then the construction nAmacBEA is a BEA scheme satisfying the above correctness and security requirement against efficient adversaries that can corrupt up to all users U_j but one and have access to the communication lines containing $mb_{i,t}, ab_{i,t}$. Furthermore, if scheme (Kg,Tag,Verify) is strongly preimage-resistant then the construction nAmacBEA satisfies security against efficient adversaries that additionally have access to the biometric storage file bsf, and to the server's secret keys.

We note that the first AMAC construction in Section 3 is weakly preimage-resistant and therefore suffices for the AMAC scheme required by Theorem 3. Furthermore, the second AMAC construction in Section 3 is strongly preimage-resistant and can therefore be used to construct the AMAC scheme required by Theorem 4.

Disclaimer. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the Army Research Laboratory or the U.S. Government.

References

1. J. Black, S. Halevi, H. Krawczyk, T. Krovetz, and P. Rogaway, *UMAC: Fast and Secure Message Authentication*, Proc. of CRYPTO '99, Springer.
2. X. Boyen, *Reusable Cryptographic Fuzzy Extractors,* Proc. of 11th ACM Conference on Computer and Communication Security, 2004
3. G. Davida, Y. Frankel, and B. Matt, *On Enabling Secure Application through Off-Line Biometric Identification,* Proc. of 1998 IEEE Symposium on Research in Security and Privacy
4. G. Di Crescenzo, R. F. Graveman, G. Arce and R. Ge, *A Formal Security Analysis of Approximate Message Authentication Codes,* Proc. of the 2003 CTA Annual Symposium, a US Dept. of Defense publication.
5. Y. Dodis, L. Reyzin, and A. Smith, *Fuzzy Extractors: How to Generate Strong Keys from Biometrics and Other Noisy Data,* Proc. of Eurocrypt 2004, Springer.
6. R. F. Graveman and K. Fu, *Approximate Message Authentication Codes,* Proc. of 3rd Annual Symposium on Advanced Telecommunications & Information Distribution Research Program (ATIRP), 1999
7. P. Indyk, R. Motwani, P. Raghavan, and S. Vempala, *Locality-Preserving Hashing in Multidimensional Spaces,* Proc. of ACM STOC 97
8. A. Jain, R. Bolle, and S. Pankanti, eds. Biometrics: Personal Identification in a Networked Society, Kluwer Academic Publishers, 1999.
9. A. Juels and M. Sudan, *A Fuzzy Vault Scheme,* Proc. of IEEE International Symposium on Information Theory, 2002
10. A. Juels and M. Wattenberg, *A Fuzzy Commitment Scheme,* Proc. of 6th ACM Conference on Computer and Communication Security, 1999
11. N. Linial and O. Sasson, *Non-Expansive Hashing,* Proc. of ACM STOC 96
12. E. Martinian, B. Chen and G. Wornell, *Information Theoretic Approach to the Authentication of Multimedia,* Proc. of SPIE Conference on Electronic Imaging, 2001
13. E. Martinian, B. Chen and G. Wornell, *On Authentication With Distortion Constraints,* Proc. of IEEE International Symposium on Information Theory, 2001
14. M. Naor and M. Yung, *Universal one-way hash functions and their cryptographic applications,* Proc. of ACM STOC 89.
15. S. Prabhakar, S. Pankanti, and A. Jain, *Biometric Recognition: Security and Privacy Concerns,* IEEE Security and Privacy Magazine, vol. 1, n. 2, March 2003.
16. B. Schneier, *Inside Risks: The Uses and Abuses of Biometrics,* Communications of the ACM, vol. 42, no. 8, pp. 136, Aug. 1999.
17. L. Xie, G. R. Arce, and R. F. Graveman, *Approximate Image Message Authentication Codes,* IEEE Transactions on Multimedia, vol. 3, June 2001.