# Codes for Interactive Authentication

Pete Gemmell* Moni Naor†

Submission to J. of Cryptology

**Abstract**

An **authentication protocol** is a procedure by which an **informant** tries to convey $n$ bits of information, which we call an **input message**, to a **recipient**. An **intruder**, $I$, controls the network over which the informant and the recipient talk. $I$ may change any message before it reaches its destination. If the protocol has security $p$, then the recipient must detect this cheating with probability at least $1 - p$.

The protocols which we consider are provably secure. In order to achieve their goal of authenticating an $n$ bit message with cheating probability at most $p$, the informant and recipient must share some small amount of secret information.

This paper is devoted to characterizing the amount of secret information that the two parties must share in a $p - secure$ protocol. We provide a single-round authentication protocol which requires $\log(n) + 5\log(\frac{1}{p})$ bits of secrecy. as well as a single-round protocol which requires $\log(n) + 2\log(\frac{1}{p})$ bits of secrecy based on non-constructive random codes. We prove a lower bound of $\log(n) + \log(\frac{1}{p})$ secret bits for single-round protocols.

We introduce authentication protocols with more than one round of communication (multi-round protocols) and present a $k$-round protocol which reduces the amount of secret information that the two parties need to $\log^{(k)}(n) + 5\log(\frac{1}{p})$. When the number of rounds is $\log^* n$, our protocol requires $2\log 1/p + O(1)$ bits. Hence interaction helps when $\log(n) > \log(\frac{1}{p})$. We also show a lower bound of $\log^{(k)} n$ on the number of shared random bits in a $k$-round protocol.

**Keywords:** Authentication, Error-Correcting Codes, Fault-Tolerance, Hashing.

*Sandia National Labs. Part of this work was done while the author was with U.C. Berkeley and with the IBM Almaden Research Center.

†Incumbent of the Morris and Rose Goldman Career Development Chair, Dept. of Applied Math and Computer Science, Weizmann Institute of Science, Rehovot 76100, Israel. Part of this work was done while the author was with the IBM Almaden Research Center. Research supported by an Alon Fellowship and a grant from the Israel Science Foundation administered by the Israeli Academy of Sciences.

# 1 Introduction

Authentication is one of the major issues in Cryptography. Authentication protocols can take on a variety of forms. The the informant and recipient may or may not rely on complexity assumptions (e.g. that factoring is hard). They may or may not wish to be able to prove to third parties that the message was indeed sent by the informant. For a general survey of authentication issues and results, the reader may refer to [9].

This paper deals with the simple scenario where two parties $A$ and $B$ communicate and want to assure that the message received by $B$ is the one sent by $A$. We provide nearly tight bounds for the case of "two party unconditionally secure authentication without secrecy" defined as follows. A protocol is "Without secrecy" if the informant and recipient make no attempt to hide the content of the input message from the intruder. In many cases the intruder may know the input message which the informant is trying to convey and wants only to convince the recipient that the informant is trying to communicate a different message.

If a protocol is "unconditionally secure," with security parameter $p$, then no intruder, regardless of computational strength, can cheat the communicating parties with probability more than $p$. An unconditionally secure protocol does not rely on complexity theoretic assumptions such as "There is no polynomial time algorithm to invert function $f$". Note that unconditionally secure protocols can be used in conjunction with computational hardness based protocols.

If we desire unconditional security then clearly the two parties must share some secret bits. In this paper we try to characterize the number of shared random bits, as a function of $p$ and $n$, where $n$ is the length of the input message, that the two parties must share in order to assure that any change made to the message will be discovered with probability at least $1 - p$. We distinguish between single-round and multi-round protocols. Single-round protocols have been investigated extensively. For this case we provide tight bounds on the number of shared bits up to constant factors: it is $\Theta(\log n + \log 1/p)$. More precisely, it is between $\log n + \log 1/p$ and $\log n + 2\log 1/p$.

In this paper we discuss the concept of multi-round authentication protocols, a subject which, to our knowledge, has not appeared in the literature. In a multi-round protocol, in order to authenticate an input message, the two parties send messages back and forth for several rounds and at the end if the (original) message has been altered it should be detected. We provide a multi-round protocol that requires $2\log 1/p + O(1)$ bits, i.e. it is independent of the message length. Hence we can conclude that interaction helps, i.e that the number shared secret bits required by a multi-round protocol is smaller than the number required by a single-round, when $\log 1/p < \log n$.

We also investigate the number of rounds required to achieve these bounds. In general, $O(\log^* n)$ round suffice to achieve the $2\log 1/p$ bound, but no constant round protocol can achieve them, since we have a lower bound of $\log^{(k)} n$ for a $k$-round protocol.

## 1.1 Previous Work

The one-round case has received a lot of attention in the literature. Gilbert, MacWilliams, and Sloane [5], who were the first to formally consider the problem, provided in 1974 a protocol requiring $2\max\{n, \log 1/p\}$ shared secret bits. Wegman and Carter [15] suggested using *$p$-almost strongly universal$_2$* hash functions to achieve authentication. They described a protocol that requires $O(\log n \log 1/p)$ secret bits.

Stinson [10] improved upon this result, using *p-almost strongly universal*$_2$ hash functions to produce a protocol which requires approximately $(2\log(n) + 3 - 2\log\log(\frac{1}{p}))(\log(\frac{1}{p}))$ secret bits. His algorithm has a lower multiplicative constant and also saves on secret bits when $\frac{1}{p}$ is large relative to $n$.

A fair amount of work has also been devoted to the question of designing protocols where the probability of cheating is exactly inversely proportional to the number of authenticators (the information sent in addition to the message) (see [3], [6], [11], [12], [13], [14]). Adding this constraint makes the task much harder. The number of secret bits required is $\Omega(n)$, and it is only possible to construct such protocols for values of $p = \frac{1}{q} : q$ a prime power.

As for lower bounds, still in the single-round case, Gilbert, MacWilliams, and Sloane [5] showed that the number of secret bits must be at least $2\log(\frac{1}{p})$, a factor of 2 higher than the obvious bound implied by the intruder simply guessing the secret bits.

Blum et al. [2] worked on the problem of checking the correctness of (untrusted) memories. They showed that a processor who wishes to store $n$ bits of information in an untrusted (adverserial) memory must have a private, trusted memory of at least $\log(n)$ bits. This lower bound argument can be converted to the authentication scenario considered in this paper.

### 1.2 Organization of the paper

In the next section we define the model and the parameters involved. Section 3 describes the single-round protocols and Section 4 the multi-round protocols. Section 5 shows the lower bounds on the number of shared random bits, both for the single-rounds and for the multi-round protocols. Section 6 shows a lower bound on the redundancy, i.e on the the length of the authenticator (the parts of the transmissions that are not the input message). Section 7 contains upper bounds for authentication series, i.e. schemes that are designed to authenticate several messages. Section 8 is a discussion on the issue of the definitions of security.

## 2 The Model

**Definition 1** *A* $k$**-round, secrecy** $l$**, probability** $p$**-authentication scheme for a message of** $n$ **bits** *is a protocol in which informant A and recipient B alternate sending each other k messages (altogether) over an insecure line controlled by an intruder I. A and B share l bits of secret information and each of them has a separate private source of random bits. Their goal is for A to communicate an arbitrary n bit input message m to B. The intruder I, which has unbounded computational power, may intercept any of their communications and replace these communications with whatever I wishes. The intruder does not have to keep A and B synchronized and can feed A with a message before B has sent it.*

*For all input messages m:*

- *When there is no interference by the adversary in the transmissions (i.e. they are all delivered unaltered), B must output m and both A and B must accept with probability at least* $1 - p$. *If, whenever there is no interference by the adversary in the transmissions (i.e. they are all delivered unaltered), B outputs m and both A and B accept with probability 1, we will call the protocol* **sound**. *We call other authentication protocols* **probabilistically sound**.

- *If $B$ receives a message $m' \neq m$, then with probability at least $1 - p$: $A$ or $B$ must output $FAIL$.*

*In the first round, $A$ sends the input message $m$ and authenticator $x_1$. In subsequent rounds $i > 1$, only an authenticator $x_i$ is sent. $A$ sends authenticators $x_1, x_3 \ldots$ and $B$ sends $x_2, x_4 \ldots$. The adversary $I$ receives each of these messages $x_i$ and replaces it with $x'_i$. $B$ receives $m', x'_1, x'_3 \ldots$ and $A$ receives $x'_2, x'_4 \ldots$. If $\exists i : x_i \neq x'_i$ then we say $I$ cheats in that round.*

If $A$ or $B$ outputs $FAIL$, then either $A$ or $B$ has detected the intruder and knows that the message delivered in the first round may not be valid. For the single-round protocols it is $B$ who detects any intrusion. For the multi-round protocols it may be either $A$ or $B$ who detects the error. Note that if we desire to have both parties alerted in the case of an intrusion, then we could add the stipulation that, at the end of the protocol, they exchange $\log \frac{1}{p}$ bit passwords which are appended to the secret string. This in particular prevents all impersonation attacks.

We note that soundness is a stronger condition than probabilistic soundness. All the protocols which we present in this paper are sound. The lower bounds which we present in sections 5.1, the secrecy of single- and multi-round protocols and for the redundancy of protocols all apply only to sound protocols. The lower bound presented in section 5.3 applies to more general probabilistically sound multi-round protocols.

### 2.1 Synchronization

For single-round protocols, synchronization is not an issue. The recipient simply waits for some authenticator, message pair to arrive and then either accepts or FAILS. For multi-round protocols, the intruder is able to carry on two separate, possibly asynchronous, conversations, one with the informant and one with the recipient. However, the party that is supposed to send the message in the $i+1$st round always waits until it receives the intruder's $i$th-round message. Therefore, for each of the two conversations, the protocol forces the intruder to commit to any possible $i$th-round cheating before soliciting the $i+1$st round message. This is used in the proof of validity for our $k$-round protocol.

Note than in a preliminary version of this paper we were not careful about the forcing the commitment. The multi-round protocol presented there is, as Gehrman [4] pointed out, susceptible to synchronization attacks. This was corrected by making *both* sides choose the random hash function used for fingerprinting the message.

## 3 Single-Round Protocols

### 3.1 $p$-Almost Strongly Universal$_2$ hash functions

For single-round protocols, Wegman and Carter [15] observed that we can view the secret shared information as a hash function, $s$, secretly chosen by $A$ and $B$ from a publicly known family of hash functions $\mathcal{H}$. If $s \in \mathcal{H}$ then $s$ maps the set $M$ of possible input messages into the set $X$ of authenticators. The requirement for the family of hash functions is that, given the value of a hash function at any one point, it must be impossible to predict the value at any other point with probability greater than $p$.

**Definition 2** *We call a hash function family $\mathcal{H}$ an* $p$**-almost strongly universal**$_2$ *if* $\forall m, m' \in M : m \neq m'$, $\forall x, y \in X, Pr_{s \in \mathcal{H}}[s(m') = x |\ s(m) = y] \leq p$.

The single-round protocols which we present are based on the following idea: $A$ and $B$ choose the secret string $s$ as a description of a member of $p$-almost-universal$_2$ family of hash functions. In order for $A$ to send $B$ the input message $m$ , it sends the authenticator pair $\langle m, s(m) \rangle$. Upon receiving the pair $\langle m', x' \rangle$, $B$ checks that $x' = s(m')$.

**Claim 1** *The probability that the intruder succeeds in fooling $B$ in the above protocol is at most $p$.*

**Proof:** From the definition of $p$-almost strongly universal$_2$ hash functions, knowing only the value of $s(m)$ for one value of $m$, $I$ can guess the value of $s(m')$, for $m' \neq m$, with probability at most $p$. □

### 3.2 A Single-Round Protocol

**Theorem 2** $\forall p > 0$, *there is a sound single-round, secrecy* $\lceil \log(n) \rceil + 5 \lceil \log(\frac{1}{p}) + 1 \rceil)$, *probability $p$ authentication scheme.*

**Proof:** The idea behind the protocol is that $A$ and $B$ share a secret hash function $s : \{0,1\}^n \rightarrow GF[Q] : Q \approx \frac{2}{p}$ chosen uniformly at random from a $p$-almost strongly universal$_2$ family of hash functions $\mathcal{H}$ such that $|\mathcal{H}| = nQ^5$. $GF[Q]$ refers to the field containing $Q$ elements. Given an input message $m$, $A$ sends $m, s(m)$ to $B$. Since $\mathcal{H}$ is $p$-almost strongly universal$_2$, $I$ has little idea what the value of $s(m')$ is for any $m'$ such that $m' \neq m$.

We now describe the construction of the $p$-almost strongly universal$_2$ hash function $s$. Let $C$ be a code $C : \{0,1\}^n \rightarrow GF[Q]^{n'}$ with the properties:

- $Q$ is equal to $\frac{2}{p}$
- $n'$ is equal to $nQ^3$
- $\forall m_1, m_2$, with $m_1 \neq m_2$, $C(m_1)$ and $C(m_2)$ differ in at least $1 - p$ fraction of their entries.

The best known construction for such a code $C$ is described by Alon et al. in [1]. The shared string $s = (i, a, b)$ consists of three random values where: $i \in_R \{1 \ldots n'\}$, $a \in_R GF[Q] - \{0\}$, $b \in_R GF[Q]$. Using those three values $s(m)$ is evaluated as: $s(m) = aC_i(m) + b$.

The single-round protocol $P_1$ is:

$P_1$: **A Single-Round, Secrecy** $\log(n) + 5\log(\frac{1}{5})$**, Probability** $p$ **Authentication Protocol**
$A$ `and` $B$ `share random secret string` $s = (i, a, b)$.
$A$: `sends to` $B$ `the pair` $\langle$ `message, authenticator` $\rangle$ `=` $\langle m, s(m) \rangle$
$B$: `receives` $\langle m', x' \rangle$ `and accepts` $m'$ `iff` $x' = s(m')$

To see that $P_1$ is a single-round probability $p$ authentication protocol, we show:

**Claim 3** *$\mathcal{H}$ is p-almost strongly universal$_2$.*

**Proof:** Fix messages $m$ and $m'$, $m \neq m'$. Let $s \in_R H, y = s(m)$. Let $x \in Q$. We will separate the analysis into two cases, $x = y$ and $x \neq y$.

1. Let $x = y$. Since $b$ is chosen uniformly at random, independent of $i$ and $a$, the distribution on $i$ given $y = s(m) = aC_i(m) + b$ is the same as the original uniform distribution on $i$. Due to the definition of the code $C$, we have: $Pr_{i \in_R [1..n']}[C_i(m) = C_i(m')] \leq p$. This implies that

$$Pr_{s \in_R \mathcal{H}}[s(m') = x | s(m) = y]$$
$$= Pr_{s \in_R \mathcal{H}}[aC_i(m') + b = s(m') = s(m) = aC_i(m) + b] \leq p$$

2. Let $x \neq y$. Choose and fix random values for $i$ and $b$. The distribution on $a$ given the knowledge $y = s(m) = aC_i(m) + b$ is the same as the original uniform distribution on $a$. Since $m' \neq m$,

$$Pr_{s \in_R \mathcal{H}}[x = s(m') | y = s(m)]$$
$$= Pr_{a \in_R GF[Q] - \{0\}}[x - y = s(m') - s(m) | y = s(m)]$$
$$= Pr_{a \in_R GF[Q] - \{0\}}[x - y = a(C_i(m') - C_i(m)) | y = s(m)] \leq \frac{1}{Q} < p$$

We have shown that $\forall m, m' : m \neq m', \forall x, y \in X$, $Pr_{s \in \mathcal{H}}[s(m') = x |\ s(m) = y] \leq p$. Therefore $\mathcal{H}$ is $p$-almost strongly universal$_2$. $\square$

### 3.3 Existence of a Single-Round, Secrecy $\log(n) + 2\log(\frac{1}{p})$ Protocol

We note here that $\forall p > 0$, there exists a sound single-round, secrecy $\lceil \log(n) \rceil + 2\lceil \log(\frac{1}{p}) + 1 \rceil)$, probability $p$ authentication scheme.

This better upper bound on the number of secret bits is attained by using a smaller family of $p$-almost strongly universal$_2$ hash functions based on a more powerful family of codes which exist, but are not necessarily constructible.

Using probabilistic arguments one can show, as was done by Roth [8], that there exists a code $C^*$ with the following properties:

- $C^*$ maps $\{0,1\}^n$ into $GF[Q]^{n'}$
- $Q$ is equal to $\frac{2}{p}$
- $n'$ is equal to $nQ^2$
- $\forall m_1, m_2$, with $m_1 \neq m_2$, and $\forall y_1, y_2 \in GF[Q]$, if $1 \leq i \leq n'$ is chosen at random, then $Pr_i[C_i^*(m_1) = y_1, C_i^*(m_2) = y_2] \leq \frac{2}{Q}$.

In this case, we could define $s = i$ where $s(m) = C_i^*(m)$.

# 4 Multi-Round Protocols

The multi-round protocols which we present in this section are based on the idea that the two parties reduce in each round the problem of authenticating the original message to that of authenticating a shorter message. The informant sends the input message in the first round and then the informant and the recipient can carry on the authentication by using a $k-1$-round protocol to authenticate a small, random "fingerprint" (defined by a hash function) of the input message that the recipient should have received. If the intruder has changed the input message that the informant sent to the recipient, then with a very high probability the fingerprint for the message received by the recipient will not match the fingerprint for the message that the informant sent in the first round. If the intruder will not alter any message sent in subsequent rounds, then the informant will be aware of the bad fingerprint sent back by the recipient.

In order to assure that the intruder is caught with the appropriate probability the two sides should choose the hash function that defines the random fingerprint

- On the fly - so as to prevent the intruder from choosing the forged message so that the true message and the forged message will collide on the hash function.
- Cooperatively - so as to prevent the intruder from using its power to synchronize the participants to learn the hash function before it should be known.

This is done using codes similar to those of the single-round protocol. The hash function is simply an index, and the value of the hash function $h$ on $x$ is the $h$th character in the codeword corresponding to $x$. The way $h$ is chosen is by letting each of the two sides $A$ and $B$ choose an index $i^A$ and $i^B$ at random. The value of $h$ is then $i^A + i^B$ (where the addition is modular). The fingerprint is then $h(x)$ concatenated with $i^A$ and $i^B$.

## 4.1 The $k$-round protocol

The protocol applies a sequence of codes $C^1, C^2, \ldots C^k$ Where $C_{j+1}$'s length is roughly logarithmic in the length of $C_j$ and the relative distance of $C_j$ is roughly $1 - \frac{p}{2^{k+2-j}}$. Let $C^j$ be a code $C^j : \{0,1\}^{n_j} \rightarrow GF[Q_j]^{\ell_j}$ where $n_j, \ell_j, Q_j$ are defined below. Let $n_1 = n$ and for $j : 1 < j \leq k$ let $n_j = \lceil \log Q_j + 2 \log \ell_j \rceil$. For $j : 1 \leq j \leq k$ The property $C^j$ should maintain is

- $\forall m_1, m_2$, with $m_1 \neq m_2$, $C^j(m_1)$ and $C^j(m_2)$ differ in at least $1 - \frac{p}{2^{k+1-j}}$ fraction of their entries.

Given the distance requirement, we should find codes where $\ell_j$ (the length of the code) and $_j$ (the alphabet size) are as small as possible. Thee Reed-Solomon codes are satisfactory; as long as $\log n_j > \frac{2^{k+2-j}}{p}$ choose $Q_j$ as a prime between $n_j$ and $2n_j$ and $\ell_j = Q_j - 1$ and $C_j$ as the corresponding RS code. The minimum relative distance of $C_j$ is at at least $1/log n_j$ which is sufficient. If $\log n_j < \frac{2^{k+2-j}}{p} < n_j$, then choose $Q_j$ as a prime between $n_j^2$ and $2n_j^2$ and $\ell_j = Q_j - 1$ and $C_j$ as the corresponding RS code. The minimum relative distance of $C_j$ is at at least $1/n_j$ which is sufficient.

We are now ready to describe the protocol. Note that the addition is mod $\ell_j$.

$P_k$: **a $k$-Round, Secrecy $\lceil log^{(k)}(n)\rceil + 5\lceil\log(\frac{1}{p})+1\rceil$, Probability $(1-\frac{1}{2^k})p$ Authentication Protocol**

$A$: send $m_1^A = m \longrightarrow B$.

$B$: receive $m_1^B$.

Do for $j = 1$ to $k-1$:

If $j$ is odd, do:

$A$: choose $i_j^A \in_R [1 \ldots \ell_j]$ and send $i_j^A \longrightarrow B$.

$B$: receive $i'_j{}^A$, choose $i_j^B \in_R [1 \ldots \ell_j]$, and send $i_j^B \longrightarrow A$.

$A$: receive $i'_j{}^B$. Compute $h_j^A = i_j^A + i'_j{}^B$ and $m_{j+1}^A = \langle i_j^A, i'_j{}^B, C^j_{h_j^A}(m_j^A)\rangle$.

$B$: compute $h_j^B = i'_j{}^A + i_j^B$ and $m_{j+1}^B = \langle i'_j{}^A, i_j^B, C^j_{h_j^B}(m_j^B)\rangle$.

If $j$ is even, do:

$B$: choose $i_j^B \in_R [1 \ldots \ell_j]$ and send $i_j^B \longrightarrow A$.

$A$: receive $i'_j{}^B$, choose $i_j^A \in_R [1 \ldots \ell_j]$, and send $i_j^A \longrightarrow B$.

$B$: receive $i'_j{}^A$. Compute $h_j^B = i'_j{}^A + i_j{}^B$ and $m_{j+1}^B = \langle i'_j{}^A, i_j{}^B, C^j_{h_j^B}(m_j^B)\rangle$.

$A$: compute $h_j^A = i_j{}^A + i'_j{}^B$ and $m_{j+1}^A = \langle i_j{}^A, i'_j{}^B, C^j_{h_j^A}(m_j^A)\rangle$.

Use protocol $P_1$, with security parameter $p/2$, to verify that $m_k^A = m_k^B$.

Note that because the two parties can combine some of their messages, the protocol is set up in such a way that it requires only $k$ rounds of communication.

An alternative way to describe the protocol is in a recursive fashion. The $k$-round protocol consists of $A$ sending the message $m$, as well as $A$ and $B$ exchanging $i_1^A$, $i_1^B$. Then the two parties authenticate the message $m_2^B = \langle i'_1{}^A, i_1^B, C^k_{i'_1{}^A + i_1^B}(m_1^B)\rangle$ using protocol $P^{k-1}$. Note that messages $m_j$ for $j > 1$ need not be sent since both $A$ and $B$ can and should compute these themselves.

**Theorem 4** *For all $n, k$ and $0 < p \leq 1$ the above protocol is a sound $k$-round, secrecy $\lceil log^{(k)}(n)\rceil + 5\lceil\log(\frac{1}{p})+1\rceil)$, probability $(1-\frac{1}{2^k})p$ authentication protocol.*

The theorem follows from the Claims 5 and 6 below.

**Claim 5** *For all $k \geq 1$, $P_k$ is a $k$ round security $(1-\frac{1}{2^k})p$ authentication scheme.*

**Proof:**

Given an execution where the above protocol fails, then $m_1^A \neq m_1^B$ (the messages sent and received are not the same) and either $m_k^A = m_k^B$ or $m_k^A \neq m_k^B$ and the failure occurred in protocol $P_1$. In case $m_k^A = m_k^B$ there must be a $1 \leq j < k$ such that $m_j^A \neq m_j^B$ but $m_{j+1}^A = m_{j+1}^B$ (call this event $D_j$). The probability of failure is then bounded by $\sum_j Pr[D_j]$ plus the probability of failure of $P_1$.

For any variable $y$ in the above protocol and for a given execution, let $T(y)$ be the time at which the variable $y$ is fixed, i.e. $T(i_j^A)$ denotes the time in which $A$ sent the message $i_j^A$ and $T(i'_j{}^A)$ is the time $B$ received the message corresponding to $i_j^A$ and $T(m_{j+1}^A) = \max\{i_j^A, i'_j{}^B\}$.

We now bound the probability of $D_j$. We will assume in the following argument that $j$ is odd. If $j$ is even, a similar argument goes through. This argument is obtained by switching the roles of $A$ and $B$ in the argument that we present here. We now assume $j$ is odd and consider two cases:

1. $T(i_j^A) < T(i_j^B)$

   We already know that:

$$\begin{aligned} T(m_j^A) &< T(i_j^A) \\ T(m_j^B) &< T(i_j^B) \end{aligned}$$

   This implies:

$$T(m_j^A), T(m_j^B) < T(h_j^B = {i'_j}^A + i_j^B)$$

   Since $T({i'_j}^A) < T(i_j^B)$, we have that $h_j^B$ is uniformly distributed in $\{1, \ldots \ell_j\}$, which from the distance property of $C^j$ implies

$$Pr[C^j_{h_j^B}(m_j^A) = C^j_{h_j^B}(m_j^B)] \leq \frac{p}{2^{k+1-j}}$$

   If $C^j_{h_j^B}(m_j^A) \neq C^j_{h_j^B}(m_j^B)$ then $m_{j+1}^A \neq m_{j+1}^B$ since either $h_j^A \neq h_j^B$ or $C^j_{h_j^A}(m_j^A) \neq C^j_{h_j^B}(m_j^B)$. Therefore

$$Pr[m_{j+1}^A = m_{j+1}^B] \leq \frac{p}{2^{k+1-j}}$$

2. $T(i_j^A) > T(i_j^B)$

   In this case

$$T(i_j^A) > T({i'_j}^A)$$

   which implies

$$Pr[i_j^A = {i'_j}^A] = \frac{1}{|m_j^A|} \leq \frac{p}{2^{k+1-j}}$$

$$\Rightarrow Pr[m_{j+1}^A = m_{j+1}^B] \leq \frac{p}{2^{k+1-j}}$$

This implies that, with probability at least $1 - p/2$, $m_k^A \neq m_k^B$. Since we run the single-round authentication protocol with security parameter equal to $p/2$, the overall probability of successful deception is at most $p$. □

**Claim 6** *$P_k$ uses $\log^{(k)}(n) + 5\log(\frac{1}{p})$ secret bits to authenticate messages of length $n$.*

**Proof:** We should show that the length of $m_k$ is small. As long as $n_j > \frac{2^{k+1-j}}{p}$ we have that $n_{j+1} = 2\log \ell_j + \log Q_j < 3 \log n_j$. We should evaluate how rapidly the length of the For $k > 1$, the number of secret bits used by $P_k$ to authenticate an $n$ bit message $m$ is the same as the number of secret bits $P_{k-1}$ uses to authenticate the message $(\langle i_k, j_k, C^k_{i_k+j_k}(m))$ which is of length $\log Q_k + 2 \log n_k$ which is at most:

$$7k + 7 + 7\log(\frac{1}{p}) + 2\log(n).$$

So long as $n^2 \geq \left(\frac{1}{p}\right)^7$, the length of the message decreases to roughly $2\log(n)$. If $n^2 < \left(\frac{1}{p}\right)^7$ then $7\log(\frac{1}{p})$ dominates other terms in the expression for the number of secret bits used. □

This concludes the proof of theorem 4. □

**Corollary 7** *For all $n$ and $p$ there exists a sound $\log^*(n)$ round, secrecy $2\log(\frac{1}{p}) + 2$, probability $p$ authentication protocol.*

**Proof:** We will use the protocol $P_{\log^*(n)}$ except that we modify the last level of recursion, using the following 1-round authentication protocol instead of $P_1$.

Consider the following single-round protocol for a message of the form $m = (x, y)$ where $x, y \in GF[Q]$. The secret string is $(a, b)$ where $a, b \in GF[Q]$. To authenticate $m = (x, y)$ send $a^2x + ay + b$. It is not hard to verify that this is a protocol for messages of length $2\log Q$, the security of this protocol is $2/Q$ and it uses a shared secret string of length $2\log Q$.

Set $p' = p/2$ and $k = \log^*(n)$ and run the protocol $P_k$ with security $p'$. When the length of the message becomes smaller than $2\log(\frac{1}{p'})$ (as it would eventually), use the above one round protocol. □

# 5 Lower Bounds

We now consider lower bounds on the number of secret bits which $A$ and $B$ require. Blum et. al. [2] showed that any single-round probability $p$ authentication protocol requires at least $\log(n)$ secret bits for any $p < \frac{1}{2}$. We improve this lower bound here.

## 5.1 Lower Bound for Sound Single-round protocols

We now show a lower bound on the number of shared secret bits in single-round protocols. The bound is achieved via a reduction from an authentication scheme to an error-correcting code. Recently, Noga Alon (private communication) improved the lower bound to $\log(n) + 2\log(\frac{1}{p}) - \log\log\frac{1}{p}$, a better lower bound, using a bound on distances for codes with maximum weight.

**Theorem 8** *There is no sound single-round, secrecy $\log(n) + \log(\frac{1}{p}) - \log\log(\frac{n}{p}) - 2$, probability $p$ authentication protocol for $p < 1$.*

**Proof:** Let $P$ be a single-round, probability $p$ authentication protocol. The outline of the proof is:

1. We define one probability distribution $\mathcal{D}_{m,x}$ on the secret strings for each input message, authenticator pair, $\langle m, x\rangle$
2. We argue that some large subset of these distributions must be "far apart".
3. We convert this subset of distributions into a set of codewords which forms a code with high minimum distance.

4. We use a lower bound from coding theory to show that the alphabet of the code (which has the same size as the set of possible secret strings) is large. Let $L$ be the number of possible secret strings. We will show that $L \log(L)$ at least $\frac{n}{p}$.

Given a fixed protocol $P$ for any message $m$ and authenticator $x$ the probability distribution $\mathcal{D}_{m,x}$ on the secret strings $s$ given that the input is $m$ and the authenticator is $x$ is well defined.

**Claim 9** *For all input messages $m$, there exists an authenticator $x$ such that for all input messages $m' : m \neq m'$ and for all authenticators $x'$, we have*

$$\sum_{s:Pr_{\mathcal{D}_{m',x'}}[s]>0} Pr_{\mathcal{D}_{m,x}}[s] \leq p$$

**Proof (of claim):**

We prove the statement by contradiction. Suppose $\exists m$ such that $\forall x \exists m' : m \neq m' \; \exists x'$ such that:

$$\sum_{s:Pr_{\mathcal{D}_{m',x'}}[s]>0} Pr_{\mathcal{D}_{m,x}}[s] > p$$

We show that with probability greater than $p$, $I$ can successfully substitute a different input message when the true input message was $m$ and therefore $P$ would not be a sound security $p$ authentication protocol.

Let the input message be $m$ and the authenticator be $x$. Let $s_0$ denote the actual secret string which $A$ and $B$ have chosen. Since $P$ is sound, $B$ must accept if there is any chance that $A$ would have sent $m', x'$ given secret string $s_0$, i.e. if $Pr_{\mathcal{D}_{m',x'}}[s_0] > 0$.

So the probability that $B$ accepts $I$'s substitution is exactly:

$$\sum_{s:Pr_{\mathcal{D}_{m',x'}}[s]>0} Pr_{\mathcal{D}_{m,x}}[s]$$

which by assumption, is greater than $p$. □

The claim implies that we can now define a set of distributions: $\{\mathcal{D}_m\}$ such that $\forall m \neq m'$,

$$\sum_{s:Pr_{\mathcal{D}_{m'}}[s]>0} Pr_{\mathcal{D}_m}[s] \leq p$$

To do so, we may set $\mathcal{D}_m = \mathcal{D}_{m,x}$ for any $x$ such that

$$\forall m', x' : m' \neq m, \sum_{s:Pr_{\mathcal{D}_{m',x'}}[s]>0} Pr_{\mathcal{D}_{m,x}}[s] \leq p$$

We now define a code which is based on these distributions.

**Claim 10** *If there is a single-round, secrecy $l$, probability $p$ authentication protocol, then there exists a code $\mathcal{C}$ with codeword length $2^l$, minimum distance at least $(1-2p)2^l$ and number of codewords $2^n$.*

**Proof (of claim):** For each distribution $\mathcal{D}_m$ on secret strings, define another distribution $\mathcal{AD}_m$, an approximation of $\mathcal{D}_m$, such that:

- Each secret string occurs in $\mathcal{AD}_m$ with probability $\frac{i}{2^l}$ where $i$ is an integer.
- $\forall s, Pr_{\mathcal{D}}[s] = 0 \Rightarrow Pr_{\mathcal{AD}}[s] = 0$
- $\forall s,\ Pr_{\mathcal{D}}[s] > 0 \Rightarrow \frac{Pr_{\mathcal{AD}}[s]}{Pr_{\mathcal{D}}[s]} \leq 2$

Define codeword $\mathcal{C}_m$ to be the lexicographically ordered multi-set which contains $i$ occurrences of $s$ for each secret string $s : Pr_{\mathcal{AD}_m}[s] = \frac{i}{2^l}$.

If two codewords, $\mathcal{C}_m$ and $\mathcal{C}_{m'}$, have distance $j$ then we have

$$\sum_{s:Pr_{\mathcal{AD}_{m'}}[s]>0} Pr_{\mathcal{AD}_m}[s] \geq (1 - \frac{j}{2^l})$$

which implies

$$\sum_{s:Pr_{\mathcal{D}_{m'}}[s]>0} Pr_{\mathcal{D}_m}[s] \geq \frac{1}{2}(1 - \frac{j}{2^l})$$

So the minimum distance distance $d$ between codewords is at least $(1-2p)2^l$. □

The dimension of $\mathcal{C}$ is $\log_{2^l}(2^n) = \frac{n}{l}$.

Any code must satisfy the inequality *dimension < length - distance + 1*, where *dimension* $= \frac{\log(\text{no. of codewords})}{\log(\text{alphabet size})}$. This implies that $\frac{n}{l} < p2^{l+1} + 1 \Rightarrow l \geq \log(\frac{n}{p}) - \log(l) - 2 \Rightarrow l \geq \log(\frac{n}{p}) - \log\log(\frac{n}{p}) - 2$.

□

## 5.2 Lower Bound for Sound $k$ Round Protocols

The idea behind our lower bound for $k$ round protocols is to show that the existence of a $k$ round, secrecy $l$, protocol implies the existence of a $k$ round, secrecy $l$, protocol whose last authenticator has at most $2^l$ bits and that the existence of this second protocol implies the existence of a $k-1$ round, secrecy $l + 2^l$ protocol.

**Definition 3** *Given a conversation consisting of input message $m$ and authenticators $x_1, x_2, \ldots x_k$, let the* **characteristic vector** *$CV(m, x_1, \ldots x_k)$ be a binary vector of length $2^l$ such that the $s$th bit, $CV(m, x_1, \ldots x_k)_s$, is 1 iff the recipient of the last message accepts given that the shared secret string was $s$ and that the conversation which the recipient of the last message saw was $m, x_1 \ldots x_k$.*

Note that, for a sound protocol, if the recipient of the last message has any chance of accepting a conversation given a particular secret string, it does so with probability 1 since it must accept all untampered conversations.

**Theorem 11** *For $p < 1$, there is no sound $k$-round, probability $p$, secrecy $\lfloor \log^{(k)}(n) \rfloor - 1)$ -authentication scheme.*

**Proof:** We will show that if there is a sound $k$-round $(p, l)$ -authentication scheme $P_k$ then there is a sound $k-1$-round $(p, l+2^l)$ -authentication scheme $P_{k-1}$.

**Claim 12** *If there is a sound $k$-round $(p, l)$ -authentication scheme $P_k$ then there is a sound $k$-round $(p, l)$ -authentication scheme $\hat{P}_k$ such that the length of the last authenticator, $x_k$, is $2^l$.*

**Proof:** Given $P_k$ we describe a protocol $\hat{P}_k$. $\hat{P}_k$ is identical to $P_k$ except for the last authenticator. The new last authenticator is the characteristic vector of the conversation that the sender of the last authenticator would have seen in $P_k$:

$$\hat{x}_k = CV(w, \ldots, x'_{k-3}, x_{k-2}, x'_{k-1}, x_k)$$

$w$ is the input message understood by the sender of the last authenticator.

The recipient of $\hat{x}'_k$ accepts iff:

- There exists an authenticator $x'_k$ such that $\hat{x}'_k = CV(w', \ldots, x_{k-1}, x'_k)$. In other words, there is an equivalent authenticator which the sender of the last authenticator could have sent in protocol $P_k$. Here $w'$ refers to the input that the recipient of the last authenticator understands.
- For the shared secret string $s$, $(\hat{x}'_k)_s = 1$. The recipient of the last authenticator would have accepted in $P_k$ if s/he received $x'_k$.

To see that $\hat{P}_k$ is a $k$-round (p,l)-authentication protocol, we show the following:

1. If $\hat{I}$, the adversary for the second protocol, does not interfere with any of the messages, then both $A$ and $B$ will accept and $B$ will know the input. This is clear since the input $m$ is sent in the first round and since the last message is $CV(m, x_1, \ldots x_k)$ where $x_1 \ldots x_k$ are the authenticators $A$ and $B$ actually send.
2. If $\hat{I}$ is able to cheat $A$ and $B$ in protocol $\hat{P}_k$ then given the same circumstances $I$ could cheat $A$ and $B$ in protocol $P_k$.

   $I$'s strategy would be to behave exactly as would $\hat{I}$ except that on the last round $I$ replaces $x_k$ with any $x'_k$ such that $\hat{x'_k} = CV(w', \ldots, x_{k-2}, x_{k-1}, x'_k)$ .

□

The proof of the theorem is now completed by defining a new $k-1$ round protocol, $P_{k-1}$:

**Claim 13** *If there exists a $k$-round $(p, l)$-authentication protocol $P_k^*$ such that the length of the last authenticator, $x_k^*$, is $|x_k^*| = 2^l$, then there exists a $k-1$ round $(p, l+2^l)$-authentication protocol, $P_{k-1}$.*

**Proof:**

**Description of $P_{k-1}$**

- We do away with the $k$th round completely by adding the advice $\overline{x_k^*}$ to the shared secret string $s$ where $\overline{x_k^*}$ is the last authenticator that would have been sent in the conversation as it would have occurred in $P_k^*$ with no interference from the adversary. The advice for the protocol $P_{k-1}$ consists of the original $l$ bits of advice from protocol $P_k$ appended to this $2^l$ bit $\overline{x_k^*}$. We note that, in this situation, the secret string depends on the input message and possibly the random bits of $A$ and $B$. However this is acceptable since the lower bound of $\log(n)$ presented in [2] applies to such protocols.
- At the end of the $k-1$st round, the party who would have sent the $k$th authenticator, $x_k^*$, in protocol $P_k^*$ instead checks to see that $x_k^* = \overline{x_k^*}$.
- The party who would have received the $k$th message checks to see that they would have accepted $\overline{x_k^*}$ in $P_k^*$ In other words, the party who would have received the $k$th message in $P_k^*$ looks at $\overline{x_k^*}$ and acts as if s/he received that.

To show that $P_{k-1}$ is a $k-1$-round $(p, l+2^l)$-authentication protocol, we note:

1. If there is been no interference by an intruder, then the party that would have sent the last authenticator in $P_k^*$ will note that $x_k^* = \overline{x_k^*}$. Furthermore, since $P_k^*$ is sound, the other party would accept $\overline{x_k^*}$.
2. If $A$ and $B$ accept an altered input message in protocol $P_{k-1}$, then the adversary in the protocol $P_k^*$ could convince $A$ and $B$ to accept by acting as s/he would in $P_{k-1}$ and then delivering, unaltered, the last authenticator $x_k^*$. The recipient of the last authenticator would accept since we have $x_k^* = \overline{x_k^*}$.

□ This concludes the proof of the theorem. □

### 5.3 Lower Bounds for Probabilistically Sound Protocols

We now consider lower bounds for protocols which are not necessarily sound: even with no interference from the adversary, they are allowed some probability of failure.

For this section, we modify the definition of characteristic vector:

**Definition 4** *$CV(m, x_1, \ldots x_k)_s = 1$ iff the recipient of the last message would accept with probability $\geq 1/2$ given the conversation $m, x_1 \ldots x_k$ and secret string $s$. Otherwise $CV(m, x_1 \ldots x_k) = 0$.*

**Corollary 14** *There is no single-round, secrecy $\log(n) - 1$, probability $p$ authentication protocol for $p < 1/3$.*

**Proof:** Suppose that $l < \log(n)$. If the secret string contains $l$ bits then there are at most $2^{2^l}$ distinct characteristic vectors. Since $l < \log(n)$ then there are fewer than $2^n$ characteristic vectors. Therefore, there is some input $m$ such that $\forall x_1 \exists x_1', m' : m' \neq m$ such that $CV(m', x_1') = CV(m, x_1)$.

The way we redefined characteristic vectors implies that the probability that $B$ will reject $m', x_1'$ is at most twice the probability that $B$ will reject $m, x_1$. Therefore, if an adversary always replaced $m, x_1$ with $m', x_1'$, with probability at least $1 - 2p > 1 - 2\frac{1}{3} = 1/3 > p$, $B$ accepts a bad message.

So we must have $p \geq 1/3$. □

**Theorem 15** *For $p < \frac{1}{3}\frac{1}{2^{k-1}}$, there is no $k$-round, secrecy $o(\log^{(k)}(n)))$, probability $p$ authentication protocol for any $c$ independent of $n$.*

**Proof:** The proof is similar to that lower bounding the number of secret bits needed in a $k$-round sound protocol. As in the previous theorem $CV(m, x_1, \ldots, x_k) = 1$ iff the recipient of the last message would accept with probability $\geq 1/2$ given the conversation has been $m, x_1 \ldots x_k$. This approximation leads to a possible doubling of the error for each conversion of the $k$ round protocol $P_k$ to a $k$ round protocol $\hat{P}_k$ which has a short last message. If the intruder $I$ has interfered in conversation $m, x_1, \ldots, x_k$ and the probability that $A$ and $B$ accept in $P_{k-1}$ is at least $q = \frac{1}{2}$ then $CV(m, x_1, \ldots, x_k) = 1$ and the probability that $A$ and $B$ accept in $\hat{P}_k$ is $1 \leq 2q$. □

# 6 Redundancy Lower Bounds for Sound Protocols

In the previous sections, we showed that multi-round protocols can be used to lessen the number of secret bits that two parties need to share in order to authenticate an $n$ bit message. However, in the protocols we presented, the number of bits exchanged, including the input message and the authenticators, was more than $n$. Here, we show a lower bound on the **redundancy**, the extra information which they have to share *or* transmit in order to authenticate an $n$ bit input message.

**Definition 5** *The* **redundancy** *of an authentication protocol is equal to the sum of the number of authentication bits – the $x_i$'s transmitted between $A$ and $B$ – plus the number of shared secret bits.*

**Theorem 16** *For any sound $k$-round authentication protocol $P$, the redundancy of $P$ is at least* $\log(n)$.

This is significant since it shows that while more rounds may decrease the number of secret bits needed, more rounds cannot decrease the redundancy below $\log(n)$.

**Proof:** Assume that the protocol $P$ uses: $t$ bits for the authenticators and $l$ bits for the shared secret string. For each input message $m$ and secret string $s$, define:

- $\mathcal{D}(m, s)$ is the probability distribution on the authenticators that would appear in a conversation between $A$ and $B$ using message $m$ and secret string $s$.
- Given a probability distribution $\mathcal{D}(m, s)$ on $t$-bit strings $\overline{x}$, the set of possible authenticator sequences for $(m, s)$ equals

$$N(\mathcal{D}(m, s)) = \{\overline{x} | Pr_{x \in \mathcal{D}(m,s)}[x] > 0\}$$

For each possible input message $m$, define a vector of sets, $V(m)$, of length $2^l$ such that $V(m)_s = N(\mathcal{D}(m,s))$.

There are $2^{2^t}$ possible subsets of all $t$ bit strings and hence at most $(2^{2^t})^{2^l} = 2^{2^{t+l}}$ possible vectors $V(m)$. If the redundancy $t+l$ is less than $\log(n)$ then the number of possible vectors $V(m)$ is less than the number of input messages.

By the pigeon hole principle, there would be two input messages, $m, m'; m \neq m'$ , which have the same vector, $V(m) = V(m')$. Because the two vectors have the same set of possible authenticator sequences in each entry, for any $s$, any authenticator sequence $\overline{x}$ which could be generated during a conversation using $m$ and $s$ could also be generated by $A$ and $B$ during a conversation using $m'$ and $s$. From soundness, we know that such authenticators must also be accepted. Therefore, if $t+l < \log(n)$, an intruder could always substitute $m'$ for $m$ with no chance of being detected. □

# 7 Authentication Series

Unlike the rest of the paper here we deal with the case where the two parties wish to communicate several unrelated messages. We discuss methods for authenticating a series of $l$ distinct $n$ bit messages where the informant wishes to send each input message before the next is known.

This topic has been previously discussed by Wegman and Carter [15] and Stinson [12]. Wegman and Carter presents a means of converting any single-round secrecy $s$ protocol into an authentication series, good for $l$ messages, using $s + l\log(1/p)$ secret bits. Stinson exhibits an authentication series, good for $l$ messages, using $(l+1)\max\{n, \log\frac{1}{p}\}$ shared secret bits.

We improve these results by slightly lowering the number of secret bits required in the single-round case and generalizing the idea of an authentication series to multi-round protocols.

We also improve Wegman and Carter's lower bound of $l\log(\frac{1}{p})$ secret bits for single-round, $l$-message authentication series and generalize Stinson's bound of $(l+1)\log(\frac{1}{p})$. The latter bound applies to the case where the probability of deception is exactly equal to 1 divided by the size of the set of authenticators. We show that any single-round series without splitting requires $(l+1)\log(\frac{1}{p})$ shared secret bits.

**Definition 6** *An* $(l,k,p,s)$**-authentication series** *for messages of* $n$ *bits is a sequence of* $l$ $k$*-round* $(p,s)$*-protocols in which players* $A$ *and* $B$ *use the same set of* $s$ *secret bits in each protocol.*

*For all possible combinations of input messages* $m^1 \ldots m^l$*, if there is no interference in any of the transmissions in any protocol, then the probability of failure in any protocol must be at most* $p$*. If there is no interference with any protocol and* $A$ *and* $B$ *accept in all protocols with probability* 1*, we say that the scheme is* **sound**.

*For all possible combinations of input messages, in the first protocol where* $B$ *does not learn the correct message,* $A$ *or* $B$ *must output* $FAIL$ *with probability at least* $1-p$ *in that protocol.*

*For the* $j$*th protocol,* $A$ *sends input message* $m^j$ *and authenticators* $x_1^j, x_3^j \ldots$ *and* $B$ *sends* $x_2^j, x_4^j \ldots$*. An adversary* $I$ *receives the input message* $m^j$ *and each of the authenticators* $x_i^j$ *and replaces them with* $m'^j$ *and* $x_i'^j$*.* $B$ *receives* $m_j', x_1'^j, x_3'^j \ldots$ *and* $A$ *receives* $x_2'^j, x_4'^j \ldots$*.*

**Theorem 17** $\forall n, \forall k : k < log^*(n)$*, and* $\forall$ *constant* $p$ *there is a sound* $(l,k,p,\lceil log^{(k)}(n)\rceil + (l+4)\lceil\log(\frac{1}{p})\rceil)$*-authentication series.*

**Proof:** Let $Q$ be a prime power equal to $\frac{1}{p}$. The secret shared information is: random $b_j \in GF[Q]$ : $j \in \{1 \ldots l\}, a_0 \in GF[Q] - \{0\}$. and $1 \leq i_0 \leq \log^{k-1}(n)$.

For each input message, $m^j$, $A$ and $B$ repeat the $k$-round $(p, \lceil \log^{(k)}(n) \rceil + 5\lceil \log(\frac{1}{p}) \rceil)$ - protocol described in theorem 4 above using $i = i_0, a = a_0, b = b_j$.

In rounds $j$ where $I$ does not cheat, $I$ gains only information of the form: $a_0 C_{i_0}(m) + b_j$. Since each $b_j$ is random and used only once, $I$ gains no information about either $a_0$ or $i_0$ and so when $I$ tries to cheat, its probability of success is the same as in the single protocol version. □

# 8 The Issue of Quantifiers

Here we address what quantifiers we will use to define authentication protocols. In both our upper and lower bound arguments, we consider protocols that have the property that **for all** messages $m$ and **for all** messages $m'$ such that $m' \neq m$, the intruder $I$ has less than probability $p$ of successfully substituting $m'$ for $m$.

This definition of authentication may seem relatively demanding on the communicating parties. There are two obvious ways in which it could be relaxed. We mention situations in which these relaxations would be inappropriate and also what effect they may have on the lower bounds.

- We could require that given a **uniform probability distribution** on the inputs message $m$, the adversary can not cheat with probability more than $p$, where the cheating probability is taken over the input message as well as the secret string and the private coins of $A$, $B$, and $I$.

  One problem with this assumption is that the actual distribution on possible inputs is not necessarily uniform. The inputs which are most likely to occur may be ones on which the adversary can successfully cheat.

  Also, the existing lower bounds on the number of secret bits apply to protocols which make this relaxation. If there is a substantial fraction of inputs on which the adversary can not cheat, then the number of secret bits required is the same as if the relaxation was not made. We believe that even protocols which may use secrecy to obscure the value of $m$ from the adversary will also require a similar number secret bits.

- If the intruder is to try to cheat on input message $m$, we require only that **for most** messages $m'$, the adversary can not substitute $m'$ for $m$ with probability more than $p$. The trouble with this approach is that, as the size of the message space increases, the number of messages $m'$ which a real-world adversary might want to substitute for $m$ may also increase. One can show that if this number grows at least as fast as $2^{\frac{n}{C}}$ for some constant $C$, then the lower bounds are unaffected.

# 9 Acknowledgments

We thank Manuel Blum for many thought-provoking and useful conversations about the defining of the problem. We thank Ronny Roth for helpful conversations about the theory of error correcting

codes. and Mike Luby for his detailed comments on the drafts of the paper. We thank Christian Gherman and Ben Smeet for fruitful discussions regarding synchronization.

## References


[1] N. Alon, J. Bruck, J. Naor, M. Naor, R. Roth, *Construction of Asymptotically Good Low-Rate Error-Correcting Codes through Pseudo-Random Graphs*, IEEE Transactions on Information Theory, Vol. 38, No. 2, March 1992

[2] M. Blum, W. Evans, P. Gemmell, S. Kannan, M. Naor *Checking the Correctness of Memories*, Proc. 31st Symp. on Foundations of Computer Science, October 1990. Full version: Algorithmica, 1994, pp. 225–244.

[3] E. F. Brickell. *A Few Results in Message Authentication* Congressus Numerantium 43 (1984), 141-154.

[4] C. Gehrman *Cryptanalysis of the Gemmell and Naor Multiround Authentication Protocol* Advances in Cryptology: CRYPTO '94, pp 121-128.

[5] E. Gilbert, F. J. MacWilliams, N. Sloane, *Codes Which Detect Deception*, The Bell System Technical Journal, Vol. 53, No. 3, March 1974

[6] M. Jimbo, R. Fuji-hara. *Optimal Authentication Systems and Combinatorial Designs,* IEEE Transactions on Information Theory, vol. 36, no 1, January 1990, pp 54-62.

[7] F. J. MacWilliams, N. Sloane. **The Theory of Error Correcting Codes,** North Holland, Amsterdam, 1977.

[8] R. Roth. Personal Communication

[9] G. Simmons, *A Survey of Information Authentication*, Proceedings of the IEEE, Vol. 76, No. 5, May 1988

[10] D. Stinson. *Universal Hashing and Authentication Codes.* Advances in Cryptology: CRYPTO '91, pp 74-85.

[11] D. Stinson. *Combinatorial Characterizations of Authentication Codes.* Advances in Cryptology: CRYPTO '91, pp 62-73.

[12] D. Stinson. *The Combinatorics of Authentication and Secrecy Codes.* Journal of Cryptology, 1990, vol.2, (no.1):23-49.

[13] D. Stinson. *Some Constructions and Bounds for Authentication Codes.* Journal of Cryptology, 1988, vol.1, 37-51.

[14] D. Stinson. *A Construction of Authentication/Secrecy Codes from Certain Combinatorial Designs* Journal of Cryptology, 1988, vol.1, (no.2):119-127.

[15] Wegman and Carter, *New Hash functions and their use in authentication and set equality* J. Computer and System Sci. **22**, 1981, pp. 265-279.