Codes for Interactive Authentication

Pete Gemmell[1] and Moni Naor[2]

[1] Sandia National Labs. Part of this work was done while the author was visiting the IBM Almaden Research Center.

[2] 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.

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 and may change any message before it reaches its destination. a If the protocol has security p, then the the recipient must detect this a cheating with probability at least $1-p$. This paper is devoted to characterizing the amount of secret information that the sender and receipient 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 $2\log(\frac{1}{p}) + O(1)$ 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.

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 [8].

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

D.R. Stinson (Ed.): Advances in Cryptology - CRYPTO '93, LNCS 773, pp. 355-367, 1994.

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, 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 $\Theta(\log n + \log 1/p)$, where n is the length of the input message. More precisely, it is between $\log n + \log 1/p$ and $\log n + 2\log 1/p$.

In this paper, we discuss 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 [4], 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 [14] suggested using ϵ-*almost strongly universal*$_2$ hash functions to achieve authentication. They described a protocol that requires $O(\log n \log 1/p)$ secret bits.

Stinson [9] improved upon this result, using ϵ-*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.

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], [5], [10], [11], [12], [13]). 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.

We apply the idea of ϵ-*almost strongly universal*$_2$ hash functions to obtain a near optimal one round protocol. One version of this protocol uses hash functions based on non-constructive codes and requires only $\log(n) + 2\log(\frac{1}{p})$ secret bits.

As for lower bounds, still in the single-round case, Gilbert, MacWilliams, and Sloane [4] 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. In 1991, Stinson [9] showed that the size of a family of ϵ-almost strongly universal hash functions, mapping a set of size a into a set of size b, is at least $\frac{a(b-1)^2}{b\epsilon(a-1)+b-a}$.

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.

We use ideas from coding theory to improve this lower bound on single-round protocols to $\log(n) + \log(\frac{1}{p})$.

The multi-round case has not been considered previously in the literature. We show that allowing k rounds of interaction between the sender and recipient enables them to get by with as few as $\log^{(k)}(n) + 5\log(\frac{1}{p})$ bits of secret information, or $\log^{(k)}(n) + 2\log(\frac{1}{p})$ bits using non-constructive codes. When $k = \log^* n$ we have a protocol requiring $2\log(\frac{1}{p}) + 2$ secreet bits. The protocols achieving these bounds reduce in every round the effective length of the message by a logarithmic factor.

We obtain a lower bound of $\log^{(k)}(n)$ for k round protocols by showing that the existence of a k round authentication protocol using l secret bits implies the existence of a $k-1$ round authentication protocol using $l + 2^l$ secret bits.

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). The full paper contains bounds for authentication series, i.e. schemes that are designed to authenticate several messages and also a discussion on the issue of the definitions of security.

Some of the lower bounds proofs are not included here and will appear in the full paper.

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:

- If there is no interference in the transmissions, then B must output m and both A and B must accept with probability at least $1 - p$.
- 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.

Definition 2. An authentication protocol P is **sound**, if, whenever there is no interference, A and B accept with probability 1.

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 ith-round message. Therefore, for each of the two conversations, the protocol forces the intruder to commit to any possible ith-round cheating before soliciting the $i+1$st round message. This is used in the proof of validity for our k-round protocol.

3 Single-Round Protocols

3.1 ϵ-Almost Strongly Universal$_2$ hash functions

For single-round protocols, Wegman and Carter [14] 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 3. We call a hash function family $\mathcal{H}$ an **ϵ-almost strongly universal$_2$** if $\forall m, m' \in M : m \neq m', \forall x, y \in X, Pr_{s \in \mathcal{H}}[s(m') = x \mid s(m) = y] \leq \epsilon$.

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 $m, s(m)$. Upon receiving the pair m', x', B checks that $x' = s(m')$.

Claim 4 *The probability that the intruder succeeds in fooling B in the above protocol is at most p.*

Proof: From the definition of ϵ-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 5. *$\forall p > 0$, 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 roughly equal to $\frac{2}{p}$
- n' is roughly 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 Sound Single-Round, Secrecy $\lceil \log(n) \rceil + 5\lceil \log(\frac{1}{p}) + 1 \rceil$, Probability p Authentication Protocol**
A and B share random secret string $s = (i, a, b)$.
A: sends to B the message, authenticator pair: $m, s(m)$
B: receives m', x' and accepts m' iff $x' = s(m')$

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

Claim 6 *$\mathcal{H}$ 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: $Pr_{i \in_r [1 \ldots n']}[C_i(m) = C_i(m')] \leq p$. 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$_2$. □

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 [7], that there exists a code C^* with the following properties:

- C^* maps $\{0,1\}^n$ into $GF[Q]^{n'}$
- Q is roughly equal to $\frac{2}{p}$
- n' is roughly 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 $Pr_i[C_i^*(m_1) = y_1, C_i^*(m_2) = y_2] \leq \frac{2}{Q^2}$.

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 informant can send the input message in the first round and then the recipient can carry on the authentication by using a $k-1$-round protocol to send back a small, random "fingerprint" of the input message it has received. If the intruder has changed the input message that the informant sent to the recipient, then with a very high probability the random fingerprint computed by the recipient will not match any 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.

4.1 The k-round protocol

The protocol applies codes similar to those used for the single-round protocol. Let C^k be a code $C^k : \{0,1\}^n \rightarrow GF[Q_k]^{n_k}$ with the properties:

- Q_k is roughly equal to $\frac{2^k}{p}$
- n_k is roughly equal to nQ_k^3
- $\forall m_1, m_2$, with $m_1 \neq m_2$, $C^k(m_1)$ and $C^k(m_2)$ differ in at least $1 - \frac{p}{2^{k-1}}$ fraction of their entries.

P_k: a k-**Round, Secrecy** $\lceil log^{(k)}(n) \rceil + 5\lceil \log(\frac{1}{p}) + 1 \rceil$, **Probability** $2(1 - \frac{1}{2^k})p$ **Authentication Protocol**
For $k = 1$ the use protocol P_1 to authenticate m.
Otherwise:
A and B share the random secret string necessary for a $(k-1)$-round protocol on inputs of size $\log(n) + 4k + 4\log(\frac{1}{p})$.
A: Send only the input message m to B
B: (after receiving m')
Choose a random index, $i_k \in_r \{1 \ldots n_k\}$.
Use the protocol P_{k-1} to send
$i_k, C_i^k(m')$ to A
A: (after the authentication of $i_k, C_{i_k}^k(m')$ is complete)
Verify that $C_{i_k}^k(m') = C_{i_k}^k(m)$
If neither party finds an inconsistency, then *ACCEPT*
Otherwise, *FAIL*

Theorem 7. *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 $2(1 - \frac{1}{2^k})p$ authentication protocol.*

Claim 8 *For all $k \geq 1$, P_k is a k round security $2(1 - \frac{1}{2^k})p$ authentication scheme.*

Proof: We have shown previously that P_1 is a single-round security $p = 2(1 - \frac{1}{2^1})p$ authentication scheme. Assume inductively that P_{k-1} is a valid $(k-1)$ round security $2(1 - \frac{1}{2^{k-1}})p$ authentication scheme.

When I commits to message m', I has no idea what the value of index i_k will be. This is true because B chooses i_k uniformly at random only after receiving m'. If $m' \neq m$ then by the definition of the code C^k, we have:

$$Pr_{i_k}[C^k_{i_k}(m) \neq C^k_{i_k}(m')] \geq 1 - \frac{p}{2^{k-1}}$$

If $C^k_{i_k}(m) \neq C^k_{i_k}(m')$ then I must cheat in the second round, i.e. not send to A the message $(i_k, C^k_{i_k}(m))$ or I will surely be caught. If I cheats in the second round, then it is caught with probability at least $1 - 2(1 - \frac{1}{2^{k-1}})p$. Therefore the probability that I can cheat successfully is at most $2(1 - \frac{1}{2^{k-1}})p + \frac{p}{2^{k-1}} = 2(1 - \frac{1}{2^k})p$. □

Claim 9 *P_k uses $\log^{(k)}(n) + 5\log(\frac{1}{p})$ secret bits to authenticate messages of length n.*

Proof: For $k > 1$, the number of secret bits used by P_k to authenticate an n bit message is the same as the number of secret bits P_{k-1} uses to authenticate a message $(i_k, C^k_{i_k}(m'))$ of length $4k + 4\log(\frac{1}{p}) + \log(n)$. So long as $n \geq \left(\frac{1}{p}\right)^4$, the length of the message decreases to roughly $\log(n)$. If $n < \left(\frac{1}{p}\right)^4$ then $5\log(\frac{1}{p})$ dominates other terms in the expression for the number of secret bits used. □

This concludes the proof of theorem 7. □

Corollary 10. *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.

Gilbert, Macwilliams and Sloane [4] showed that any single-round probability p authentication protocol requires at least $2\log(\frac{1}{p})$ secret bits. Their argument was based on lower bounding $H(s)$, the entropy of the secret string s. They

showed that $H(s) = H(sx|m) = H(x|m) + H(s|mx)$, where $H(X|Y)$ is the conditional entropy of X given Y, averaged over all possible Y's.

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 second 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.

Theorem 11. *There exists a function f such that $f(x) = o(\log(x))$ and such that there is no sound single-round, secrecy $\log(n) + \log(\frac{1}{p}) - f(\frac{n}{p})$, 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, (m, x).
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 $\frac{n}{p}$.

The rest of the proof appears in the full paper. 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.

5.2 Lower Bound for 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 12. 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 sth 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 13. *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 14 *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 15 *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 kth 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 kth authenticator, x_k^*, in protocol P_k^* instead checks to see that $x_k^* = \overline{x_k^*}$.
- The party who would have received the kth 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 kth 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, because 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 because we have $x_k^* = \overline{x_k^*}$.

□ This concludes the proof of the theorem. □

5.3 Lower bounds for protocols which are not necessarily sound

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 16. $CV(m, x_1, \ldots x_k)_s = 1$ 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 17. *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 18. *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

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 19. 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 20. *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 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. Also, we thank Mike Luby for his detailed comments on the drafts of the paper.

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.
3. E. F. Brickell. *A Few Results in Message Authentication* Congressus Numerantium 43 (1984), 141-154.
4. E. Gilbert, F. J. MacWilliams, N. Sloane, *Codes Which Detect Deception*, The Bell System Technical Journal, Vol. 53, No. 3, March 1974
5. 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.
6. F. J. MacWilliams, N. Sloane. **The Theory of Error Correcting Codes**, North Holland, Amsterdam, 1977.
7. R. Roth. Personal Communication
8. G. Simmons, *A Survey of Information Authentication*, Proceedings of the IEEE, Vol. 76, No. 5, May 1988
9. D. Stinson. *Universal Hashing and Authentication Codes.* Advances in Cryptology: CRYPTO '91, pp 74-85.
10. D. Stinson. *Combinatorial Characterizations of Authentication Codes.* Advances in Cryptology: CRYPTO '91, pp 62-73.
11. D. Stinson. *The Combinatorics of Authentication and Secrecy Codes.* Journal of Cryptology, 1990, vol.2, (no.1):23-49.
12. D. Stinson. *Some Constructions and Bounds for Authentication Codes.* Journal of Cryptology, 1988, vol.1, 37-51.
13. D. Stinson. *A Construction of Authentication/Secrecy Codes from Certain Combinatorial Designs* Journal of Cryptology, 1988, vol.1, (no.2):119-127.
14. Wegman and Carter, *New Hash functions and their use in authentication and set equality* J. Computer and System Sci. **22**, 1981, pp. 265-279.