J. Cryptology (1990) 2: 77–104

Journal of Cryptology

A Cartesian Product Construction for Unconditionally Secure Authentication Codes that Permit Arbitration[1]

Gustavus J. Simmons
Sandia National Laboratories,
Albuquerque, NM 87185, U.S.A.

Abstract. An authentication code consists of a collection of encoding rules associating states of an information source with messages that are to be used to communicate the state to a designated receiver. In order for a collection of encoding rules to be useful as an authentication code there must also exist one or more probability distributions on the rules which, if used by the receiver and transmitter (the insiders) to choose secretly the encoding rule they use, will result in the receiver being able to (probably) detect fraudulent messages sent by an outsider or modifications by him of legitimate messages.

Authentication codes that permit arbitration are codes that in addition to protecting the insiders from deception by outsiders, also protect against some forms of insider deception. This is accomplished by making it possible for an arbiter to resolve (again in probability) certain disputes between the transmitter and receiver: the transmitter disavowing a message that he actually sent or the receiver claiming to have received a message that the transmitter did not send.

An infinite class of authentication codes that permit arbitration is constructed and some bounds on the probability of a deception going undetected are proven. These codes are shown to be unconditionally secure, i.e., it is shown that the probability of a deception either going undetected or else of being unjustly attributed to an innocent party is independent of the computing capability or investment that a would-be cheater is willing to make.

Key words. Authentication, Authentication codes, Arbitration, Unconditional security.

1. Introduction

The object of authentication has traditionally been to protect against two types of deception that can occur in a communications channel. In this context authentication has been restricted to providing the authorized receiver(s) with a capability of detecting unauthentic messages, i.e., either messages originated by unauthorized transmitters, or else authentic messages that have been intercepted and either replaced or modified before being relayed on to the receiver. Brickell [2], De Soete [6], Massey [12], Schoebi [17], Simmons [18]–[20], and Stinson [26], [27] have

[1] Date received: May 1, 1987. Date revised: January 15, 1990. This work was performed at the Sandia National Laboratories and was supported by the U.S. Department of Energy under Contract No. DE-AC04-76DP00789.

constructed unconditionally secure authentication codes that realize this limited objective, but only subject to the severe constraint that the transmitter and receiver have had to be assumed to act with the joint purpose of detecting attempted deceptions by outsiders and that neither would attempt to deceive the other. This assumption was made necessary by the fact that, prior to the discovery of the authentication codes described in this paper, the only unconditionally secure codes that had been devised required both the transmitter and receiver to know the same secret (from the opponent) information, and hence for them to have interchangeable capabilities either to generate or verify authentic messages. The unavoidable consequence has been that the known codes have all left open the possibility for either the transmitter to disavow a message that he actually sent to the receiver or else for the receiver to attribute falsely a message of his own devising to the transmitter. Of course the party being deceived would realize that he was the victim of a deception by the other, but would be unable to prove this to a third party. Ideally, authentication should provide a means to protect against deceptions by insiders (the transmitter or receiver) as well as by outsiders (the opponent). It has been an open question until recently [3], [22], [23] of whether it was even possible to devise unconditionally secure authentication codes that would permit a fourth party, an arbiter, to decide (in probability) whether the transmitter or the receiver was cheating in the event of a dispute. We answer this question in the affirmative by first constructing an example of an unconditionally secure Cartesian product authentication code that, while still permitting the receiver to detect outsider deceptions, also makes it possible for a predesignated arbiter to corroborate (in probability) insider deceptions. The construction in the example code is then generalized to an infinite class of Cartesian product codes that also permit arbitration. These codes are unconditionally secure in the sense that the probability of a deception going undetected is independent of the computing power or time that the party attempting a deception may bring to bear. This is in contrast to either computationally secure codes where the security depends on a would-be cheater having to carry out some computation that in principle is possible but in which all of the known methods of execution require an infeasible amount of computation or provably secure codes where it can be shown that the security is at least as great as the difficulty of carrying out some related (presumed) hard problem, such as factoring suitably chosen large composite integers, etc.[2] It should be pointed out, however, that for this particular construction of authentication with arbitration codes, the requirement that the transmitter and receiver unconditionally trust each other has, unfortunately, only been replaced by a requirement that they each unconditionally trust the arbiter. Recently Brickell and Stinson [3] have extended the notion of authentication codes that permit arbitration reported here and devised a protocol which permits arbitration with multiple arbiters that avoids the necessity for unconditionally trustworthy arbiters—at the expense of requiring a much larger exchange of private keying information to set up the authentication code and a corresponding increase

[2] This distinction between unconditionally secure schemes (not necessarily restricted to authentication codes) and provably secure schemes is of relatively recent origin and is made necessary by a growing body of results of each type. In [24], for example, Simmons classified as provably secure, authentication codes that under currently accepted usage of the term would be called unconditionally secure.

in the information content of the messages used to communicate the states of the source.

Functionally, there is little difference between computationally secure or provably secure and unconditionally secure systems. The difference is in how the functions are realized. Although this paper is concerned exclusively with unconditionally secure codes, we remark briefly on the essential features that underlie both types of authentication schemes. Clearly there must be some operation on the information conveyed in an authenticated message which is computationally feasible for the authorized transmitter to perform, but which is either improbable or else computationally infeasible for an opponent to do, and whose correctness of execution can be verified by at least the authorized receiver(s), and perhaps by the arbiter(s).

Since an opponent can always choose a message using whatever strategy he wishes and communicate it to the receiver on the chance that it might be accepted as an authentic communication from the legitimate transmitter, for authentication to be possible at all, it must be true that only a subset out of the collection of all possible messages will be acceptable to the receiver as authentic at any given time. The essential concept on which authentication is based irrespective of whether the scheme is only computationally or provably secure or is unconditionally secure, is to organize the sets of acceptable messages in such a way that no matter which set the receiver has chosen, the opponent's probability of finding a message in that set will be small enough for the resulting security against deception to be acceptable. Any message chosen by the opponent, using whatever strategy he elects to use, should have some probability of being rejected as unauthentic since it may be one of the messages that are unacceptable to the receiver, and which, therefore, would not have been sent by the legitimate transmitter.

In computationally and provably secure authentication schemes, the sets of acceptable messages are often determined (virtually constructed) by either appending a cryptographically related message authentication code (MAC) to the information being authenticated, or else by first appending an unrelated authenticator and then cryptographically "sealing" the resulting extended message using either a single key or a two-key cryptoalgorithm. Each choice of a key defines one subset of acceptable messages. In the case of authentication schemes based on single-key cryptography this is unavoidable since the only operation that the insiders (who know the key) can do, that outsiders cannot, is to encrypt or decrypt information using the secret key. In the case of two-key cryptographic techniques though, or especially in the case of a pure authentication channel, this need not be true. This is because an authentication channel can differ significantly from a secrecy channel, since in the one case it is only necessary that the receiver be able to verify that the authentication operation has been correctly carried out in order to establish that the communication is authentic, while in the other case he must be able to "invert" the operation to recover the information actually concealed in the cipher. A well-known example of the latter type, i.e., of using a public-key encryption algorithm to define the set of acceptable messages by concealing them in ciphers is the digital signature scheme defined by Rivest *et al.* [16]. In this case, the information being authenticated is concealed by the authentication operation and revealed as an essential part of the process of verifying its authenticity. It is, of course, essential

that only a predesignated fraction of the messages will be accepted as authentic: for example, those ending in a pre-agreed suffix. On the other hand, the digital signature schme of El Gamal [7] is essentially an appended authenticator (MAC) to the message which need not be, and in fact cannot be, decrypted by the receiver in the process of verifying that the message is authentic. Both of these schemes are computationally secure as are most of the other digital signature schemes based on two-key cryptographic techniques. There are, however, a few provably secure authentication (digital signature) schemes [9], [15], [25], [28] based on public-key cryptographic algorithms. The digital signature scheme of Goldwasser *et al.* [9] in addition passes a very strong security requirement, namely, it is secure against adaptive chosen message attacks. This is an appropriate point to remark that a digital signature is more than just a computationally secure or provably secure authentication with arbitration scheme. Anyone (having access to entirely public information) can verify the authenticity of a signature—not just the predesignated arbitor(s). It is an important point and should be clearly stated that the price paid to achieve unconditional security (in all presently known realizations) is to restrict the ability to authenticate messages to insiders, i.e, to parties possessing some information not known to all of the other participants.

In all cases in which encryption, either single-key or two-key, is used to define the set of acceptable messages, the authenticated messages (ciphers) consist of only those ciphers that decrypt with the chosen key to either a meaningful message or to a message with the proper appended authenticator or else they consist of messages in which encryption is used to form an appended authenticator. The coincidence of acceptable messages between subsets determined by different keys is, in general, computationally infeasible to determine so that only statistical statements can be made about the opponent's likelihood of being able to deceive the receiver. For example, even today after a decade of open community research, no plaintext–ciphertext pair is known which is fixed by two or more nonweak Data Encryption Standard (DES) keys although such pairs almost certainly exist.[3] In other words, we do not know of any nontrivial overlap of plaintext–ciphertext pairs between acceptable message sets defined by different DES keys. If a single-key cryptoalgorithm is used then the transmitter and receiver must both know the key; the transmitter so that he can encrypt (authenticate) information and the receiver so that he can decrypt or verify the authenticity of the messages he receives. This is true irrespective of whether the information being authenticated is known to, or kept secret from, the opponent, i.e., independent of whether the authentication is made with or without secrecy [21]. In either case, if a single-key cryptoalgorithm is being used, the transmitter and the receiver must both keep the key secret (from the opponent) as well as ensuring its integrity (against substitution or modification). However, if a two-key (née public-key) algorithm is used, the receiver need only ensure the authenticity, i.e., the integrity, of the key that he uses, not its secrecy. The transmitter must, of course, keep secret and protect the integrity of the key he uses to encrypt, i.e., to authenticate, messages. Using encryption to define the set of

[3] Note added in revision: This comment is no longer true. Quisquater and Delescaille [14] have announced (February 1989) that they have found several nonweak DES collisions.

acceptable messages has been the classic way in which communications have been authenticated, by military and diplomatic services for example, especially where the transmitter and receiver were mutually trusting and trustworthy so that they could share a single cryptographic key which each could trust the other to keep secret and to not misuse to deceive him. Such a scheme is only as secure as the cryptoalgorithm is difficult to cryptanalyze, which is measured by the computational difficulty of carrying out the calculations necessary to cryptanalyze the system.

2. The Threat

Since the function of authentication codes is to protect against deceptions involving unauthentic information, we start by describing the three types of deception that one or more of the known codes can detect and/or in probability prevent. It should be pointed out, though, that there are many other types of deceptions besides those considered here. For example, one of the participants in an information-based protocol may deliberately reveal information, which, according to the protocol, he is supposed to keep secret, thereby impeaching the integrity of the protocol and hence of the information itself, or a cabal of the participants may pool their private pieces of information in an attempt to cheat one or more of the other participants, etc. In the broadest sense, the subject of authentication codes includes providing protection against all forms of deceit where the success of the deception depends on causing one or more of the participants in an information-based protocol to accept unauthentic information and to act as though it were authentic. However, for the purposes of this paper, we restrict attention to only three generic types of deception, two of which have further natural subdivisions depending on the information the would-be cheater has available to him at the time he tries to cheat.

The terminology which we use in discussing authentication is well established in the literature, but unfortunately suggests a narrower view of the subject than is intended. Since the problem of authenticating information first arose in a communications context, the convention is to call the authorized originator of the authenticated information the transmitter, and the authenticated information which he originates, a message. This message, devoid of any meaningful physical embodiment, is communicated to a remotely located receiver over a publicly exposed, noiseless, communication channel, which is commonly referred to as the authentication channel. In the simplest possible authentication scheme, the intended recipient of the message, the authorized receiver(s), is also the party desiring to verify its authenticity; although, as the discussion in the next few paragraphs will make clear, there are circumstances in which this is not the case. Authentication, however, is much broader than this communications-based terminology would suggest. The information to be authenticated may indeed be a message in a communications channel, but it could equally well be data in a computer file or resident software; it could quite literally be a fingerprint in the application of the authentication channel to the verification of the identity of an individual [13], [19] or figuratively a "fingerprint" in the verification of the identity of a physical object such as a document or a tamper-sensing container [11], etc. In the broadest sense, authen-

tication is concerned with establishing the integrity of information purely on the basis of the internal structure of the information itself, irrespective of the source of the information.

In the most general model of authentication there are four essential participants, the "insiders" are the transmitter (the authorized originator for messages), the legitimate receiver(s), and, depending on the particular authentication scheme being used, perhaps the arbiter(s). Whether the arbiter is an insider or an outsider depends on whether he is in possession of any privileged information, i.e., information not available to one or more of the other participants. The fourth participant, the opponent, is always an outsider who is assumed to have no privileged information, but who is assumed to be knowledgeable of the general authentication scheme being used by the transmitter and receiver (an extension of Kerckhoff's criteria in cryptology to authentication) and to be capable of sophisticated eavesdropping, computation, and message alterations. Given this general setting, there are (at least) four classes of cheating (attempted deceptions) that can occur: classified by the identity of the would-be cheater.

The opponent can send a fraudulent message to the receiver in hopes of having it accepted as an authentic communication from the transmitter. He can do this after having eavesdropped on l, $l \geq 0$, legitimate messages communicated to the receiver by the transmitter. We denote this type of cheating by the notation $\mathbf{I}_0$ or $\mathbf{I}_l$ and $\mathbf{S}_l$, $l \geq 1$. The cases $\mathbf{I}_0$ and $\mathbf{I}_l$, $l \geq 1$, are sufficiently different that we describe them separately:

$\mathbf{I}_0$: The opponent, based only on his knowledge of the general authentication scheme being used by transmitter and receiver can send a fraudulent message to the receiver when in fact no message has yet been sent by the transmitter. The probability of his succeeding in deceiving the receiver in this case is simply the value of the two-person game whose representation is the incidence matrix of the authenticating rules—mapping source information into messages—in the general authentication scheme [18], [20], [21]. This is an easy computation to carry out, even for large authentication schemes. $\mathbf{I}_0$ is commonly referred to as (the opponent) impersonating the transmitter.

$\mathbf{I}_l$ and $\mathbf{S}_l$: The opponent can wait to observe $l - 1$ legitimate messages from the transmitter which he allows to pass to the receiver without tampering with them. When he intercepts the lth message there are two courses of action available to him: he can either substitute some other message of his own devising in its stead or he can forward it without modification to the receiver. He could then, based on what he has learned from the l observations he has made of legitimate messages, send a message of his own choosing to the receiver, i.e., he can attempt to impersonate the legitimate transmitter. The first type of deception is an lth-order substitution attack, $\mathbf{S}_l$, where $\mathbf{S}_1$ is commonly referred to as simply substitution, while the second type of deception is an lth-order impersonation. The opponent's strategy in this case, $l \geq 1$, is defined by conditional probabilities, i.e., his decision of which message to substitute will be affected by the legitimate messages he has observed (and also by whether he knows or does not know the information

being conveyed by the observed messages [18], [20], [21]). Already for the case of $\mathbf{S}_1$ this problem is computationally difficult (even for modest-sized authentication schemes) unless very stringent conditions are imposed on the regularity and symmetry of the associated designs.

Simmons has restricted attention in his earlier work on authentication codes [18], [20]–[23] and in this paper to deceits $\mathbf{I}_0$ and $\mathbf{S}_1$, i.e., transmitter impersonation and/or message substitution. The reason for this is that, for opponent deceits with $l > 1$, an *ad hoc* rule must be introduced to prevent the opponent from simply substituting a legitimate message already observed prior to the lth communication and hence known to be acceptable to the receiver, for the lth message—if they are different. Various other authors [6], [12], [17], [27] have considered authentication codes for the cases $l > 1$. In all cases, the opponent wins if the receiver accepts the fraudulent message as being an authentic communication from the transmitter *and*, if $l > 0$, ends up being misinformed as to the transmitter's communicated information in consequence.

Insider cheating involves a participant who knows some piece of information about the authentication scheme not known to all of the other participants: the transmitter, receiver, or in some instances as noted earlier, the arbiter(s). We do not consider the case of insider-arbiter cheating since the authentication codes to be described here provide no protection against this type of deceit. In the most general setting, though, arbiter cheating is a fourth type of deception that needs to be protected against in addition to the three considered here. Protection against transmitter or receiver cheating presupposes that there is an arbiter who will arbitrate disputes between them, i.e., who will assign liability to the party most likely to be responsible. This arbiter, for the scheme described in this paper to work, must be assumed to be unconditionally trustworthy. In other words, we assume that the arbiter will not misuse his privileged position to deceive either the transmitter or receiver.

The receiver can cheat if he can successfully attribute a message of his own devising to the transmitter, i.e., a message not sent by the transmitter. "Successfully" means that when the transmitter later claims (correctly) that he did not send the message in question, that the arbiter will rule against him. The receiver can wait to attribute a fraudulent message to the transmitter until after he has received l, $l \geq 0$, legitimate messages. We denote this second form of cheating by the notation $\mathbf{R}_l$, $l \geq 0$.

$\mathbf{R}_l$: The receiver, using both the public knowledge of the general authentication scheme and his privileged information, claims to have received from the transmitter, in an authentic message, fraudulent information of his own devising. He is successful if and only if the arbiter later certifies the fraudulent message as being one that the transmitter could have sent under the existing protocol. If he attempts to cheat before any legitimate message has been sent, this is an $\mathbf{R}_0$ deception. If he waits until after he has received l legitimate messages, $l \geq 1$, from the transmitter, and then—using this additional information—attempts to cheat in the same way as before, this is said to be an $\mathbf{R}_l$ deception.

The category $\mathbf{R}_l$, $l > 1$, will not be considered here for the same reason that the categories $\mathbf{I}_l$ and $\mathbf{S}_l$, $l > 1$, were excluded from consideration for opponent deceptions.

T: The transmitter can attempt to cheat the receiver by sending a message which the receiver may accept and then claiming that he did not send it, i.e., by disavowing a legitimate message. He will be successful in this deceit if and only if when the receiver later claims to have been cheated the arbiter rules that the message is not one that the transmitter would have sent under the existing protocol and, in consequence, the receiver is held liable.

An arbiter can attempt to deceive the receiver in precisely the same ways that an opponent can, using his privileged position and information, if any. In the scheme to be described here, the arbiter would be certain of success. However, in the general case, or perhaps even for other arbitration protocols, an arbiter would not necessarily be certain of either having a fraudulent message of his choosing accepted by the receiver or, if it was, of having it attributed in the arbitration protocol to the transmitter. The arbiter's options are (like the opponent's) either to impersonate the transmitter after observing $l \geq 0$ legitimate messages, or else to substitute a fraudulent message for a legitimate one on the lth round, $l \geq 1$. If the arbiter is in possession of some privileged information, i.e., is an insider, so that he can be distinguished from the opponent who was defined to be an outsider, then we would propose using the notation.

$$I_l^a \quad \text{and} \quad S_l^a$$

to indicate the various types of arbiter deceptions. No further mention or use of arbiter deceptions will be made in this paper.

It is perhaps useful in understanding insider deceits to think of the receiver as a stockbroker and the transmitter as one of the broker's customers. In this setting it is easy to believe that a customer might wish to disavow an order that he actually issued if it later turns out that the decision was a bad one that cost him money. Similarly, the broker, who is managing the customer's account, might very well wish to execute an order of his own devising when he had received no such instructions from the customer, or even to execute orders contrary to the customer's instructions, to generate commissions for himself or in his judgment to make better investments. In either case, the function of authentication would be, in the event of a dispute between the broker and the customer as to whether the broker had faithfully carried out the customer's instructions, to make it possible for an impartial third party to decide who was, in all probability, liable.

3. A Partial Solution: Unconditionally Secure A-Codes

Pedagogically the easiest way to approach authentication with arbitration, $\mathbf{A}^2$-codes, is to do so in two steps: first, discussing very briefly the essential features of the conceptually simpler authentication codes that do not permit arbitration, A-codes, and then extending these to codes that make arbitration possible. A-codes provide protection only against opponent deceits and, as already remarked, for the purposes of this paper, only against deceptions of types $\mathbf{I}_0$ and $\mathbf{S}_1$.

In the case of unconditionally secure authentication codes, A-codes, the code consists of the explicit specification of the collection of encoding rules (mapping source information into acceptable messages) which the transmitter and receiver can choose from and the associated protocol that the transmitter and receiver will jointly (and secretly) select one of these rules to specify the subset of messages which the transmitter will use and which the receiver will accept (as authentic). Given such an explicit construction, Simmons has shown [18], [20], [21] that it is possible to compute the precise value for the probability that an opponent will be able to deceive the receiver, even if he is assumed to have unlimited computing power and time. In practice this computation is often infeasible to carry out even for quite modest-sized codes unless very strong constraints are imposed on the incidence structure of the sets of acceptable messages [2], [20], [21], [24], [26].

In the most general model for unconditionally secure authentication codes there is a source $\mathcal{S}$ whose state, s, is the information that the transmitter wishes to convey to the (legitimate) receiver(s). There is also a probability distribution S on the states of $\mathcal{S}$ which we assume cannot be influenced by any of the participants. The number of states of $\mathcal{S}$, each of which is assumed to have a nonzero probability of occurrence, is σ; $|\mathcal{S}| = \sigma$. There is also a noiseless, but insecure, communications channel which can communicate any message, m_i, out of a set $\mathcal{M}$; $|\mathcal{M}| = v$. The transmitter/receiver choose $\mathcal{M}$ as a part of their design of an authentication code. The probability distribution, M, of the usage (by the transmitter) of messages from $\mathcal{M}$ to communicate states of the source to the intended receiver is a complex function of the authentication code, as well as of the actions of the various participants. There are practical considerations, such as the most efficient use of the channel in which $\mathcal{M}$ occurs and even more importantly of the most efficient use of the (costly) secure channel needed to set up the authentication channel, which dominate the transmitter/receiver's choice of an authentication code. We ignore these considerations here, and simply analyze one class of authentication codes, assuming that they have been chosen using whatever criteria are relevant.

Clearly, since the whole object is for the transmitter to be able to inform the receiver of the state of the source, there must be at least as many messages as there are source states

$$v = |\mathcal{M}| \geq |\mathcal{S}| = \sigma. \tag{1}$$

Equally clearly, strict inequality must hold in (1), otherwise all messages would have to be usable (by the transmitter) and hence be acceptable (to the receiver), and the opponent could impersonate the transmitter with certainty of success using any message whatsoever. With no loss of generality, this says that in any particular instance, a proper subset, $\mathcal{M}_i \subset \mathcal{M}$, of the messages will be used, i.e., available for the transmitter to use and acceptable to the receiver. Again, since there must be at least one message available to convey each state of the source $|\mathcal{M}_i| \geq \sigma$. The transmitter and receiver must have a common understanding of the relationship between messages and states of the source; the transmitter so that he can choose a message, m, communicating the observed state, s, (encoding) and the receiver so that he can interpret m to infer that s occurred. This says that in addition to the choice of a proper subset, $\mathcal{M}_i \subset \mathcal{M}$, $|\mathcal{M}_i| \geq \sigma$, a mapping of $\mathcal{S}$ onto $\mathcal{M}_i$ must also be specified by the transmitter/receiver. Each state of the source must be mapped onto

at least one message in $\mathcal{M}_i$ and perhaps more than one. Since we assume, in accordance with Kerckhoff's criteria in cryptography, that the opponent is fully knowledgeable of the authentication code, i.e., that he knows the family of subsets, $\mathcal{M}_i$, and the associated mappings of $\mathcal{S}$ onto them that the transmitter/receiver can choose among. The coincidence of messages between these subsets must satisfy a number of conditions for authentication to be possible. For example, it should be obvious that no message can occur in all of the subsets, otherwise the opponent could impersonate the transmitter using that message with certainty of success and every message must occur in some subset, otherwise that message could be deleted from $\mathcal{M}$ without affecting the performance of the code. In the least restricted formulation of A-codes, any given message may be used in different encoding rules to communicate different states of the source. An encoding rule can be thought of as a labeling of a proper subset of the messages with the states of the source in which each state must be used to label at least one message. However, if each message always conveys the same source state (in all of the encoding rules in which it is used) the code is said to be Cartesian since the labeling in the individual rules is that induced by a labeling with the σ states of the source of the parts in a partition of $\mathcal{M}$ into σ nonempty parts.

In general, what is done to make it feasible to solve for the probability of the receiver being deceived (into accepting an unauthentic message as authentic) is to insist that all encoding rules use the same number of messages, that each message occur in the same number of encoding rules, and—if secrecy is not required—that the code be Cartesian.

To simplify the description of the authentication codes to be discussed here, we define a matrix product $\otimes$ reminiscent of a Cartesian product for sets and which we therefore refer to as a Cartesian product (of matrices). For matrices M_1 and M_2 with $r_1(r_2)$ rows and $c_1(c_2)$ columns, respectively, we define $\mathbf{A} = M_1 \otimes M_2$ to be the matrix with $r_1 r_2$ rows and $c_1 + c_2$ columns such that, for $1 \le i \le r_1$ and $1 \le j \le r_2$, the $(r_2(i-1)+j)$th row of $\mathbf{A}$ is the concatenation of the ith row of M_1 with the jth row of M_2. The extension to the product of more than two factor matrices is obvious. We often use the notation

$$\mathbf{A} = A \otimes A \otimes \cdots \otimes A = A^{\sigma}$$

to denote an A-code for a $|\mathcal{S}| = \sigma$ state source in which the messages in the ith factor are all labeled with source state σ_i. We refer to an A-code constructed in this way as a Cartesian product code.

The smallest possible example of a Cartesian product A-code which can be used to communicate one bit of information about the state of the source (two equally likely states, say H and T) and provide one bit of authentication has $e = m = 4$:

$$A = \begin{pmatrix} \mathrm{H} & - \\ - & \mathrm{H} \end{pmatrix} \otimes \begin{pmatrix} \mathrm{T} & - \\ - & \mathrm{T} \end{pmatrix} = \begin{array}{c|cccc} & m_1 & m_2 & m_3 & m_4 \\ \hline a_1 & \mathrm{H} & - & \mathrm{T} & - \\ a_2 & \mathrm{H} & - & - & \mathrm{T} \\ a_3 & - & \mathrm{H} & \mathrm{T} & - \\ a_4 & - & \mathrm{H} & - & \mathrm{T} \end{array} . \tag{2}$$

Only a moment's reflection is needed to see that the authentication scheme in (2) does indeed provide one bit of authentication protection against both $\mathbf{I}_0$- and

S_1-type deceptions. It is worth pointing out why this true, since these same principles are also the basis for the construction of authentication codes that permit authentication. First the Cartesian product construction ensures an independence between the probability of a successful deception and the probability distribution on the source states. To see this, assume that the source is strongly biased so that $P(H) \gg P(T)$, i.e., so that m_1 or m_2 are correspondingly much more likely to be seen as messages than m_3 or m_4. This clearly has no effect on the opponent's probability of success in an I_0 deceit. In an S_1 deception, if the opponent observes m_1 or m_2—which is likely—he is still forced with a choice between m_3 or m_4 as the substitute message, either of which is equally likely to be acceptable (or unacceptable) to the receiver. The same argument holds if he observes m_3 or m_4. Thus as a consequence of the Cartesian product construction his probability of deceiving the receiver will be 1/2, i.e., one-bit of uncertainty for either an I_0- or an S_1-type deception irrespective of the probability distribution on the source states.

The fact that **A** can be represented in this example as a Cartesian product of two 2×2 labeled matrices and that it is also a Cartesian authentication scheme may mislead the reader into believing that the two properties are synonymous. What is true is that while a Cartesian product code is always Cartesian (due to the one-to-one association of the factors in the product with states of the source) the converse does not hold in general. The authentication code

$$A = \begin{array}{c|cccc} & m_1 & m_2 & m_3 & m_4 \\ \hline a_1 & \mathrm{H} & - & \mathrm{T} & - \\ a_2 & \mathrm{H} & - & - & \mathrm{T} \\ a_3 & \mathrm{H} & \mathrm{H} & \mathrm{T} & - \\ a_4 & - & \mathrm{H} & - & \mathrm{T} \end{array}$$

is Cartesian but **A** is clearly not representable as a Cartesian product of factor matrices.

4. A Small Example of an Unconditionally Secure A^2-Code

The smallest possible example of an A^2-code which provides one bit of protection against all five types of deception, while communicating one bit of information about the source, say the outcome of a fair coin toss, is an extension of the one-bit source Cartesian product construction of an **A**-code described above.

The essential notion to constructing A^2-codes is that each potential cheater must be uncertain as to the choice of messages that he can successfully cheat with, which means that if the code is to provide one-bit of protection against all forms of deception, then each participant must be faced with a choice between two messages each of which is equally likely to succeed (or fail) in the intended deception no matter what he does. From the standpoint of the opponent this is no different than the design criteria for **A**-codes. However, for the insiders, this means that the transmitter must be uncertain as to which messages the receiver will accept, but at the same time he must know at least one acceptable message that can be used to communicate each possible state of the source. The inescapable conclusion is that it must be the

case that the receiver will accept more messages (as authentic) than those that the transmitter knows to be acceptable. Conversely, the receiver must be uncertain of which messages the transmitter knows to be acceptable and hence will use, otherwise he could, with certainty of success, attribute a message to the transmitter when none had been sent. The receiver must, therefore, be uncertain about the subset of messages that the transmitter will legitimately use, out of the set of messages that he (the receiver) will accept, even after learning one message from that set through a legitimate communication from the transmitter.

In analogy to A-codes where we specified both the set of encoding rules and the protocol for their use, $\mathbf{A}^2$-codes also consist of both the specification of a set of authentication/encoding rules and of a protocol for their use. Even though we often refer to the matrix representation of the set of rules as the $\mathbf{A}^2$-code, the reader should keep in mind that a protocol for their usage is also a part of the code. To construct the one-bit example of an $\mathbf{A}^2$-code, the twofold Cartesian product in (2) of factors of the form $\begin{pmatrix} s_i & - \\ - & s_i \end{pmatrix}$, where s_i is one of two possible states for the source, is replaced by the twofold product

$$A = \begin{bmatrix} \mathrm{H} & \mathrm{H} & - & - \\ \mathrm{H} & - & \mathrm{H} & - \\ - & \mathrm{H} & - & \mathrm{H} \\ - & - & \mathrm{H} & \mathrm{H} \end{bmatrix} \otimes \begin{bmatrix} \mathrm{T} & \mathrm{T} & - & - \\ \mathrm{T} & - & \mathrm{T} & - \\ - & \mathrm{T} & - & \mathrm{T} \\ - & - & \mathrm{T} & \mathrm{T} \end{bmatrix} \tag{3}$$

or

$$A = \begin{array}{c|cccccccc} & m_1 & m_2 & m_3 & m_4 & m_5 & m_6 & m_7 & m_8 \\ a_1 & \mathrm{H} & \mathrm{H} & - & - & \mathrm{T} & \mathrm{T} & - & - \\ a_2 & \mathrm{H} & \mathrm{H} & - & - & \mathrm{T} & - & \mathrm{T} & - \\ \vdots & \vdots & & & & \vdots & & & \vdots \\ a_{15} & - & - & \mathrm{H} & \mathrm{H} & - & \mathrm{T} & - & \mathrm{T} \\ a_{16} & - & - & \mathrm{H} & \mathrm{H} & - & - & \mathrm{T} & \mathrm{T} \end{array} \tag{4}$$

A, which is the authentication scheme, is assumed to be known to everyone: the transmitter, the receiver, the opponent, and the arbiter in an extension of Kerckhoff's criteria in cryptography to authentication theory.

The authentication with arbitration protocol, in this example of an $\mathbf{A}^2$-code, calls for the receiver to choose one out of the 16 authenticating rules, $\mathbf{a}_i$, that make up **A**. For example, $\mathbf{a}_1$ says that the receiver will accept messages m_1 and m_2 as authentic and will interpret either message to mean that the outcome of the coin toss was heads (H). Similarly, messages m_5 or m_6 would be accepted and interpreted to mean that the outcome of the coin toss was tails (T), while messages m_3, m_4, m_7, and m_8 would be rejected by the receiver as unauthentic. The important point to note is that in each of the authenticating rules there are exactly two acceptable (to the receiver) messages available to communicate each state of the source. The receiver informs the arbiter in secret (from the transmitter and the opponent(s)) of his choice of an authenticating rule. According to the protocol, the receiver commits himself to accept as authentic precisely those four messages corresponding to the source states appearing in the authenticating rule he chose and to reject the remaining four

as unauthentic. There is no conceivable reason for the receiver to violate his agreement and to accept or claim to have received a message not used in the authenticating rule he selected, since it could not communicate a state of the source and would assuredly not be certified by the aribiter in any event.

The protocol calls for the arbiter to construct an encoding rule for the transmitter to use by selecting a subset of the messages appearing in **a** such that there is at least one message which can be used to communicate each state of the source. For this small example, the arbiter has only one option, namely to choose one out of each pair of messages that the receiver agreed to accept as conveying each of the source states. Since the receiver knows that this is what the arbiter will do, the arbiter must choose between the messages in each pair with a uniform probability distribution in order to keep the receiver one bit uncertain as to which message the transmitter will be told he can legitimately use. One mechanism the arbiter can use to do this, which generalizes in a nice way for this particular class of Cartesian product constructions, is to choose one of the four vectors defined by the product

$$\Pi = \begin{pmatrix} 1 & - & - & 1 \\ - & 1 & 1 & - \end{pmatrix} \otimes \begin{pmatrix} 1 & - & - & 1 \\ - & 1 & 1 & - \end{pmatrix}, \tag{5}$$

again with a uniform probability distribution, and form the Schur product[4] of the vector he chooses with the encoding rule selected by the receiver. The net result is that one out of the 16 possible encoding rules

$$E = \begin{array}{c|cccccccc} & m_1 & m_2 & m_3 & m_4 & m_5 & m_6 & m_7 & m_8 \\ a_1 & \mathrm{H} & - & - & - & \mathrm{T} & - & - & - \\ a_2 & \mathrm{H} & - & - & - & - & \mathrm{T} & - & - \\ \vdots & \vdots & & & & \vdots & & & \vdots \\ a_{15} & - & - & - & \mathrm{H} & - & - & \mathrm{T} & - \\ a_{16} & - & - & - & \mathrm{H} & - & - & - & \mathrm{T} \end{array} \tag{6}$$

will be selected as a result of the concatenated choices of the receiver and arbiter. There are other ways to achieve the same end result but, as we shall see later, it is easy to explain why this procedure succeeds. According to the protocol, the arbiter is committed to certify as messages that could have been used by the transmitter (under the protocol) only those that correspond to source states appearing in the encoding rule (Schur product) he has constructed. The arbiter communicates, in secret (from the receiver and the opponent(s)), this encoding rule to the transmitter. The transmitter is supposed to use this rule to encode an observed state of the source into the message he will transmit to the receiver. Assume, for example, that the receiver chose authenticating rule, $\mathbf{a}_1$, and that the arbiter chose the vector

$$- \ 1 \ 1 \ - \ 1 \ - \ - \ 1,$$

then the resulting encoding rule would be

$$\mathbf{e} \begin{array}{|cccccccc|} \hline m_1 & m_2 & m_3 & m_4 & m_5 & m_6 & m_7 & m_8 \\ - & \mathrm{H} & - & - & \mathrm{T} & - & - & - \\ \hline \end{array}. \tag{7}$$

[4] Given vectors $\mathbf{A} = (\mathbf{a}_i)$ and $\mathbf{B} = (\mathbf{b}_i)$, the Schur product, $\mathbf{C} = \mathbf{A} \odot \mathbf{B}$, is the vector $\mathbf{C} = (\mathbf{a}_i\mathbf{b}_i)$.

Using this encoding rule, the transmitter would communicate source state H by sending message m_2 while T would be communicated by sending message m_5. According to the protocol, messages m_2 and m_5 would not only be accepted as "authentic" by the receiver, but would also, in the event of a later dispute between the transmitter and the receiver as to the authenticity of a message, be certified by the arbiter as messages the transmitter could have sent. Of course, the receiver would also accept m_1 and m_6 as authentic, however, the transmitter does not know this and the arbiter would not certify either of these messages as ones the transmitter would have sent under the established protocol.

The protection afforded by this authentication scheme against each of the five types of cheating described earlier, holds the cheater to a probability of success of 1/2, i.e., one bit of protection irrespective of which type of deception is considered. The easiest to analyze are the $\mathbf{I}_0$ and $\mathbf{S}_1$ deceptions since that is a game between only the opponent and the receiver. The opponent knows the authentication scheme, i.e., he knows **A** and the authentication protocol but does not know the receiver's or arbiter's choices. It should be clear that the receiver can limit the opponent's chances of success in either an $\mathbf{I}_0$ or an $\mathbf{S}_1$ deception to 1/2 by choosing an authenticating rule from **A** using a uniform probability distribution on the rows. If the receiver's only concern was to protect himself against deception by the opponent, he could choose rows from among four different subsets of the rows of **A** each containing only four rows—each a redundant representation (in messages) of the A-code constructed in (2):

$$\begin{pmatrix} \mathrm{H} & \mathrm{H} & - & - \\ - & - & \mathrm{H} & \mathrm{H} \end{pmatrix} \otimes \begin{pmatrix} \mathrm{T} & \mathrm{T} & - & - \\ - & - & \mathrm{T} & \mathrm{T} \end{pmatrix}$$

or

$$\begin{pmatrix} \mathrm{H} & \mathrm{H} & - & - \\ - & - & \mathrm{H} & \mathrm{H} \end{pmatrix} \otimes \begin{pmatrix} \mathrm{T} & - & \mathrm{T} & - \\ - & \mathrm{T} & - & \mathrm{T} \end{pmatrix}$$

or

$$\begin{pmatrix} \mathrm{H} & - & \mathrm{H} & - \\ - & \mathrm{H} & - & \mathrm{H} \end{pmatrix} \otimes \begin{pmatrix} \mathrm{T} & \mathrm{T} & - & - \\ - & - & \mathrm{T} & \mathrm{T} \end{pmatrix}$$

or

$$\begin{pmatrix} \mathrm{H} & - & \mathrm{H} & - \\ - & \mathrm{H} & - & \mathrm{H} \end{pmatrix} \otimes \begin{pmatrix} \mathrm{T} & - & \mathrm{T} & - \\ - & \mathrm{T} & - & \mathrm{T} \end{pmatrix}.$$

His optimal strategy in either of these four cases (against the opponent) would be to choose one of four rows with a uniform probability distribution which would limit the opponent to a probability of success of 1/2 in either an $\mathbf{I}_0$- or $\mathbf{S}_1$-type deception. The transmitter, however, would be certain of success in a **T** deception since in all of the four-row subarrays, the identification of a message (to the transmitter by the arbiter) that is to be used to communicate a state of the source, unambiguously identifies the other message that the receiver would accept as communicating that same state. Consequently, the only scheme available to the receiver that will limit the probability of his being deceived by the opponent to 1/2 and not allow the transmitter to disavow messages with a certainty of success is **A**,

and an optimal strategy for the receiver in this case is the uniform probability distribution on the rows of **A**. If the opponent attempts an $\mathbf{O}_0$ deception and sends a message when none has yet been sent by the authorized transmitter, his probability of choosing one of the four (out of eight) messages that the receiver has agreed to accept is 1/2, since, for each encoding rule, there are four equally likely messages that will be accepted as authentic and four that will be rejected as unauthentic. On the other hand, if he waits to observe a message his uncertainty about the encoding rule chosen by the receiver drops from one out of 16 equally likely candidates to one out of four; however, irrespective of the message he observes, the four encoding rules including that message will leave him with four equally likely possibilities for the message that the transmitter would use to communicate the other state of the source. More importantly, he is also faced with four equally likely pairings of messages that the receiver would accept as communicating the other state of the source, with each message occurring in precisely two of the pairs. The net result is that the opponent's probability of success in an $\mathbf{O}_1$ deception is also 1/2.

Consider next the case of the transmitter attempting to disavow a message that he actually sent, i.e., a **T** deception. In order for him to succeed, he must choose a message that is used in the authenticating rule that the receiver chose from among the rows of **A**, but not the one used in the encoding rule constructed by the arbiter. Continuing with the example given above, the transmitter can infer, from the encoding rule communicated to him by the arbiter,

$$- \ \mathrm{H} \ - \ - \ \mathrm{T} \ - \ - \ -,$$

that the receiver must have chosen one of the four authenticating rules in which m_2 and m_5 appear:

m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8
H	H	–	–	T	T	–	–
H	H	–	–	T	–	T	–
–	H	–	H	T	T	–	–
–	H	–	H	T	–	T	–

(8)

Since messages m_3 and m_8 do not appear in any of these rules, the transmitter can be certain that they would be rejected by the receiver as unauthentic, and hence he will certainly not use either of these. Each of the remaining four messages, m_1, m_4, m_6, and m_7, appears in two out of the four equally likely authenticating rules. Therefore, he cannot do better than to randomly choose one out of these four messages with a uniform probability distribution. Irrespective of which of the four he chooses, the probability that it will be accepted by the receiver is 1/2. If it is accepted, the transmitter can cheat and disavow having sent it, since he knows that the arbiter will not certify it as a message that he would have used under the established protocol. Precisely the same situation would hold for any other pair of choices by the receiver and arbiter, i.e., for all choices of an encoding rule.

Finally, we consider the two types of cheating available to the receiver, $\mathbf{R}_0$ and $\mathbf{R}_1$ deceptions. Of the four messages that the receiver has agreed to accept as authentic, two (one for each state of the source) will be certified by the arbiter as being messages that could have been used by the transmitter under the established

protocol since they appear in the encoding rule and two will not be certified. The receiver will succeed in cheating, i.e., fraudulently attributing a message to the transmitter, only if he chooses one out of the two that the arbiter will certify and will fail otherwise. It should be clear that his probability of success is 1/2 since the arbiter's procedure for generating an encoding rule chose among the acceptable (to the receiver) messages with a uniform probability distribution. If the receiver waits until he receives a message from the transmitter, say m_2, he can reduce his uncertainty about the vector that the arbiter used in constructing the encoding rule from one out of four equally likely cases to one out of two:

$$- \ 1 \ 1 \ - \ 1 \ - \ - \ 1$$

or

$$- \ 1 \ 1 \ - \ - \ 1 \ 1 \ -$$

in the example, but his uncertainty about the message chosen by the arbiter to convey the other state of the source is still one out of two equally likely cases, m_5 and m_6. Hence his probability of successfully substituting a message conveying a different state of the source than was communicated in the message sent by the transmitter (an $\mathbf{R}_1$ deception) is also 1/2. By successful, we mean that the substitute message will be certified by the arbiter to be one that the transmitter could have sent since it appears in the encoding rule constructed by the arbiter.

This example illustrates all of the essential features of unconditionally secure authentication codes that permit arbitration: $\mathbf{A}^2$-codes. Three bits of information must be transmitted to specify one of eight equally likely messages. This provides one bit of information (to the receiver) about the source state, one bit of protection (to the transmitter and receiver) against $\mathbf{I}_0$ and $\mathbf{S}_1$ deception by outsiders, and one bit of protection (to either the transmitter or else to the receiver) against cheating by insiders, i.e., $\mathbf{R}_0$ and $\mathbf{R}_1$ on $\mathbf{T}$ deceptions. Since the probability of success for the "cheater" in all cases is the same as the probability that a randomly chosen message from the set of possible (according to what he knows of the scheme) messages would be successful in achieving the objectives of the cheater, it seems reasonable to describe the code described here as perfect in direct analogy to the usage of the term in connection with A-codes.

5. A Cartesian Product Construction for Unconditionally Secure $\mathbf{A}^2$-codes

In generalizing the example of an $\mathbf{A}^2$-code constructed in the previous section, there is an obvious question as to which properties in the example are essential to such codes, and which are merely artifacts of that example. We do not attempt to answer this question, but instead simply point out the properties that we have chosen to preserve in the $\mathbf{A}^2$-codes that we construct, whether essential to $\mathbf{A}^2$-codes in general or not.

First, the protocol will be the same in the general case as for the example. The receiver will choose an authenticating rule, **a**, from among the rows of an array **A** representable as a Cartesian product of identical factor arrays A and then communicate **a** in secret (from the transmitter and opponent(s)) to the arbiter. The receiver is committed to accept as authentic only those messages appearing in **a**. As

mentioned earlier, there is no conceivable motive (to the receiver) to violate this commitment. The arbiter constructs an encoding rule, **e**, which is a submapping from **a** of S into M, i.e., only messages used in **a** can appear in **e**, and each state of the source must appear at least once in **e**. The arbiter then communicates **e** in secret (from the receiver and opponent(s)) to the transmitter. The arbiter is essentially "playing" against the receiver at this point since the receiver is free to choose any row of **A** as the authenticating rule **a** from among whose elements the arbiter must construct an encoding rule **e**. We assume therefore that the arbiter's strategy will be to construct **e** so as to minimize the receiver's probability of deception. This must leave available to the receiver a strategy (for choosing the **a**) that will minimize both the probability of the transmitter and the opponent being able to deceive him: P_T and P_O. In general (for arbitrary **A**), these may be inconsistent goals, but for the construction of Cartesian codes to be described here they are consistent. The transmitter is supposed to use **e** to encode an observed state of the source into a message for communication to the receiver. The arbiter's primary function is to certify later whether a disputed message, m, is one that appeared in the encoding rule **e** which he constructed, i.e., whether m is a message that the transmitter could conceivably have sent when acting in accordance with the established protocol. This basic protocol would be the same for all $\mathbf{A}^2$-codes, irrespective of what their structure might be.

We want it to be the case that an optimal strategy for the receiver to use in selecting an authenticating rule will be to choose a row from **A** with a uniform probability distribution. We also want it to be true that an optimal strategy for the arbiter to use in constructing the encoding rule, **e**, will be to choose a fixed number $\alpha \geq 1$ of elements from among the k elements appearing in each factor of **A**, again with a uniform probability distribution over the k elements. We say that an authentication code is uniform if the uniform probability distribution on the possibilities is an optimal strategy for both of the parties who must make a choice. In the case of **A**-codes where only the receiver/transmitter make a choice—which can best be thought of as the receiver making a choice of an encoding rule which he then secretly communicates to the transmitter—there are infinitely many such uniform codes [2], [8], [18], [20], [21], [26] as well as infinitely many nonuniform codes [1], [20], [21], [26]. In this section we construct one infinite class of uniform $\mathbf{A}^2$-codes and conjecture that there are also infinitely may more codes, both uniform and nonuniform, although no example of the latter category has been constructed to date.

In order to simplify the calculation of the level of authentication provided by an $\mathbf{A}^2$-code we restrict attention to a class of Cartesian product constructions that generalize the example of the preceding section.

A two-level pairwise balanced design (PBD) with parameters

$$\text{PBD}(v, b, k, r, \{\lambda, 0\})$$

is a design with v elements and b blocks, each block contains k elements and each element occurs in r blocks and each pair either occurring in λ blocks or in no blocks at all [10]. In the example above A was a

$$\text{PBD}(4, 4, 2, 2, \{1, 0\}),$$

but in general we will let the factor A be any $\text{PBD}(v, b, k, r, \{\lambda, 0\})$ design.

We define an $\mathbf{A}^2$-code to be equitable if the probability of success for all types of deception are the same, i.e., if

$$P_{\mathbf{O}_0} = P_{\mathbf{O}_1} = P_{\mathbf{R}_0} = P_{\mathbf{R}_1} = P_{\mathbf{T}}. \tag{9}$$

Theorem 1. *Given a factor array A defined by a two-level pairwise balanced design*

$$\text{PBD}(v, b, k, r, \{\lambda, 0\})$$

the $\mathbf{A}^2$*-code defined on the* σ*-fold Cartesian product*

$$\mathbf{A} = A \otimes A \otimes \cdots \otimes A = A^{\sigma}$$

with $\alpha = 1$, *i.e., in which the arbiter chooses a unique message from each factor for the transmitter to use in communicating the corresponding state of the source, has the following properties*:

1. *The protection provided by the code against all types of deception is independent of the probability distribution on the source states.*
2. *The probability of attempted deceptions being successful will be*

$$P_{\mathbf{I}_0} = P_{\mathbf{S}_1} = P_{\mathbf{O}} = \frac{r}{b} = \frac{k}{v},$$

$$P_{\mathbf{R}_0} = P_{\mathbf{R}_1} = P_{\mathbf{R}} = \frac{1}{k}, \tag{10}$$

$$P_{\mathbf{T}} = \frac{\lambda}{r}.$$

3. *The code is uniform*
4. *If, in addition,* $v = k^2$ *and* $\lambda = r/k$, *then the code is equitable and* $P_{\mathbf{D}} = 1/k$ *for all deceptions* **D**.

Proof. Two properties of the $\mathbf{A}^2$-codes constructed here follow directly from the Cartesian product construction of **A**. First we point out that the observation of a message communicating an arbitrary source state cannot help in a deception involving a different source state. In particular, this says that, for the opponent, $P_{\mathbf{I}_0} = P_{\mathbf{S}_1} = P_{\mathbf{O}}$ and that, for the receiver, $P_{\mathbf{R}_0} = P_{\mathbf{R}_1} = P_{\mathbf{R}}$. The second property is that the probability of any type of deception ($\mathbf{I}_0$, $\mathbf{S}_1$, $\mathbf{R}_0$, $\mathbf{R}_1$, or **T**) being successful is independent of the probability distribution on the source states, S. To see this, let **D** be an arbitrary deception and let $P_{\mathbf{D}}(s_i)$ be the probability that **D** will be successful given that s_i is the source state. Clearly,

$$P_{\mathbf{D}}(s_i) = P_{\mathbf{D}}(s_j) = P_{\mathbf{D}}$$

for all i and j by the symmetry of the Cartesian product construction. Hence the value to a would-be cheater of attempting deceit **D** will be

$$\sum_{s_i \in \mathscr{S}} P(S = s_i) P_{\mathbf{D}}(s_i) = P_{\mathbf{D}}.$$

It should be noted that this property (of the security being independent of the source

state distribution) is a consequence of the Cartesian product construction of **A** and *does not* hold for authentication codes in general.

We analyze the probability of a successful deception separately for each of the three potential cheaters.

The presence of the arbiter, and even the fact that the code is an $\mathbf{A}^2$-code and not simply an **A**-code, has nothing to do with the opponent's strategy or with his chances of deceiving the receiver. From his viewpoint, he is merely playing an impersonation game against the receiver (since observing a message occurring in one of the factors leaves him with the problem of impersonating the transmitter in one of the other factors in an $\mathbf{S}_1$ deception). We have proven earlier[5] (Theorem 1 of [20]) that the probability of a successful impersonation $P_{\mathbf{I}}$ is bounded by

$$P_{\mathbf{I}} \geq \frac{k}{v} \tag{11}$$

for an arbitrary authentication code. The receiver can ensure that the opponent can do no better than (11) by using a uniform probability distribution in choosing the row of **A** that specifies the messages he will accept as authentic. One optimal strategy for the opponent in this case, i.e., ensuring that equality holds in (11), would be to choose a message with a uniform probability distribution: from among all σv messages for an $\mathbf{I}_0$ deception, or from among the $(\sigma - 1)v$ messages occurring in factors other than the one in which the observed message occurs in an $\mathbf{S}_1$ deception. In either case

$$P_{\mathbf{I}_0} = P_{\mathbf{S}_1} = P_{\mathbf{O}} = \frac{k}{v} = \frac{r}{b}. \tag{12}$$

The receiver can also be viewed as playing purely a substitution game against the arbiter—but on a different authentication code. The remarks made earlier about the independence of the factors in the Cartesian product apply in this case as well, so that we can restrict attention to a single factor. Although there are v messages appearing in each factor, the receiver has identified (to the arbiter) k of those that he will accept as authentic, one of which the arbiter has chosen to be the one the transmitter is to use. The receiver and arbiter are therefore playing an impersonation game on an **A**-code with a $k \times k$ array having a single entry in each row and column. The theorem cited above says that the receiver's probability of a successful deception in this case is bounded by

$$P_{\mathbf{R}} \geq \frac{1}{k}. \tag{13}$$

The same arguments used to analyze the opponent's case apply here as well. Consequently, the arbiter can ensure, by using a uniform probability distribution to choose one out of the k acceptable messages, that the receiver can do no better than the bound of $1/k$ while the receiver can ensure that he achieves this value by impersonating with a uniform probability distribution on the possible messages.

---

[5] What is proven there is that $P_{\mathbf{I}_0} \geq \min_i |\mathbf{a}_i|/|\mathcal{M}|$, but in the present case $\min_i |\mathbf{a}_i| = |\mathbf{a}_i| = \sigma k$ for all i and $|\mathcal{M}| = \sigma v$, so that $P_{\mathbf{I}_0} \geq k/v$.

Therefore,

$$P_{\mathbf{R}_0} = P_{\mathbf{R}_1} = P_{\mathbf{R}} = \frac{1}{k}. \tag{14}$$

The transmitter's chances of success can also be calculated as the value of a game between him and the receiver/arbiter. As before, the factors of **A** are independent and can be treated independently. The joint result of the receiver's and the arbiter's actions is to reveal to the transmitter a message appearing in the encoding rule chosen by the receiver. The transmitter can cheat if and only if he can choose another message appearing in that same rule, in which case the receiver will accept the message as authentic but the arbiter will not certify it as a message the transmitter would have used. This is simply a substitution game on A—without splitting. In other words, any message other than the one (for each source state) that the transmitter is supposed to use that the receiver will accept (as authentic) will be a win for the transmitter.

The transmitter, upon being told that he is to use a particular message to encode a given state of the source, will know that the authenticating rule chosen by the receiver was one out of the set of r rules in which the specified message occurs. In each of these there are $k - 1$ other messages that would also be acceptable to the receiver. The receiver's strategy for selecting authenticating rules will have determined the probability of any given set of $k - 1$ of the messages being acceptable. Let X be the set of all messages that occur in some authenticating rule with a particular message m_0. Since m_0 occurs in r rows of A, each row of which contains k entries, there are $r(k - 1)$ occurrences of other messages paired with m_0 over all of the rows of A. Since A was chosen to be a two-level pairwise balanced design with pairs occurring either λ times or else not at all, any message that occurs with m_0 in some row of A will also occur the same number of times, λ, as any other message that occurs with m_0. The number of distinct messages that occur with m_0 is therefore

$$|X| = \frac{r(k-1)}{\lambda},$$

$k - 1$ of which occur with m_0 in the row of A that the receiver chose and communicated to the arbiter. Clearly, if the transmitter picks one out of these $|X|$ messages with the uniform probability distribution, the probability that it will be acceptable to the receiver (i.e., that the transmitter will succeed in deceiving the receiver) is λ/r, independent of the choice of the arbiter. Therefore,

$$P_{\mathbf{I}} \geq \frac{\lambda}{r}. \tag{15}$$

Conversely, if the arbiter and the receiver both make choices with the uniform probability distribution, then, for each $m \in X$, the probability that the arbiter will certify m is also λ/r. Therefore,

$$P_{\mathbf{I}} = \frac{\lambda}{r} \tag{16}$$

in this case.

We have shown that if the receiver and arbiter both use the uniform probability distribution in making their choices, then the probability of success for all types of deception are as small as is possible. Consequently, the Cartesian product $\mathbf{A}^2$-codes described here are uniform.

The last result of the theorem is an immediate consequence of replacing v by k^2 and λ by r/k in (12), (14), and (16) to get

$$P_{\mathbf{O}} = P_{\mathbf{R}} = P_{\mathbf{T}} = \frac{1}{k}, \tag{17}$$

which shows that the code is equitable in this case. □

In the small (one-bit source) example we used the 4×4 array A:

$$\begin{array}{cc|cccc} & a_1 & 1 & - & 1 & - \\ \pi_1 & a_4 & - & 1 & - & 1 \\ \hline & a_2 & 1 & - & - & 1 \\ \pi_2 & a_3 & - & 1 & 1 & - \end{array}, \tag{*}$$

where $r = 2$, $g = 1$, and $k = 2$, with the rows rearranged for reasons that will be apparent in a moment. We have not labeled the columns with the messages associated with them as we did earlier for the A-codes since each factor involves v dedicated messages for the communication of a single state of the source. From (17) we know that v must be a square for the code to be equitable, therefore the next smallest array must have $k = 3$ and $v = 9$. One such array is the following:

$$\begin{array}{cc|ccccccccc} & a_1 & 1 & 1 & 1 & - & - & - & - & - & - \\ \pi_1 & a_2 & - & - & - & 1 & - & - & 1 & 1 & - \\ & a_3 & - & - & - & - & 1 & 1 & - & - & 1 \\ \hline & a_4 & 1 & - & - & 1 & 1 & - & - & - & - \\ \pi_2 & a_5 & - & 1 & - & - & - & 1 & - & 1 & - \\ & a_6 & - & - & 1 & - & - & - & 1 & - & 1 \\ \hline & a_7 & 1 & - & - & - & - & 1 & 1 & - & - \\ \pi_3 & a_8 & - & 1 & - & 1 & - & - & - & - & 1 \\ & a_9 & - & - & 1 & - & 1 & - & - & 1 & - \end{array}, \tag{**}$$

where $r = 3$, $\lambda = 1$, and $k = 3$.

The reader may have recognized that the two incidence arrays (*) and (**) for $k = 2$ and 3 and $\lambda = 1$ are examples of what Bruck [4], [5] has defined to be a finite net. A finite net (i-net), $\mathbf{N}$, is a set of $v = k$ elements (points) and ik k-subsets of points (lines) which can be grouped into i parallel classes of k lines each, such that distinct lines of the same parallel class have no points in common, while any two lines from different classes have exactly one point in common. Finite nets are closely

related to finite affine planes. A finite net can sometimes be embedded in an affine plane, however, every subset of i spreads of parallel lines from an affine plane is necessarily an i-net. It is this result that we use to construct an infinite family of $\mathbf{A}^2$-codes. The arrays (*) and (**) were in fact constructed in just this way by deleting an arbitrary spread of lines from a resolution of the affine planes with $v = 2$ and $v = 3$. The general means of constructing perfect authentication codes that permit arbitration can now be stated:

Theorem 2. *Uniform, equitable Cartesian* $\mathbf{A}^2$*-codes exist for all* $k = p^\beta$, p *a prime, with the parameters*

$$v = b = k^2,$$

$$r = k,$$

$$\lambda = 1,$$

and

$$P_D = \frac{1}{k}.$$

Proof. Start with a finite affine plane with $v = k^2$ points (whose existence is assured for all $k = p^\beta$, p a prime) and form a k-net by choosing any subset of k parallel spreads of lines out of the $k + 1$ spreads in all. Identify lines of the resulting k-net with authenticating rules to form the factor array A. Then A defines a PBD(k^2, k^2, k, k, $\{1, 0\}$). The result follows from Theorem 1. □

It is worth noting that other factor arrays, A, can be formed from a particular affine plane by choosing i, $i \neq k$, of the parallel spreads. For example in the affine plane $EG(2, 2)$ corresponding to the case $k = 2$, there are three parallel spreads (or parallel resolutions of the associated design)

$$\begin{array}{ll} \pi_1 & (12)(34) \\ \pi_2 & (13)(24) \\ \pi_3 & (14)(23) \end{array}$$

and three nonisomorphic A arrays corresponding to choosing one, two, or all three of these spreads:

$$\pi_1 \begin{array}{|cccc|} \hline 1 & 1 & - & - \\ - & - & 1 & 1 \\ \hline \end{array},$$

$i = 1$,

$$\begin{array}{cc} \pi_1 & \begin{array}{|cccc|} \hline 1 & 1 & - & - \\ - & - & 1 & 1 \\ \hline \end{array} \\ \pi_2 & \begin{array}{|cccc|} \hline 1 & - & 1 & - \\ - & 1 & - & 1 \\ \hline \end{array} \end{array},$$

$i = 2$,

or

$$\begin{array}{cc} \pi_1 & \begin{array}{|cccc|} \hline 1 & 1 & - & - \\ - & - & 1 & 1 \\ \hline \end{array} \\ \pi_2 & \begin{array}{|cccc|} \hline 1 & - & 1 & - \\ - & 1 & - & 1 \\ \hline \end{array} \\ \pi_3 & \begin{array}{|cccc|} \hline 1 & - & - & 1 \\ - & 1 & 1 & - \\ \hline \end{array} \end{array},$$

$i = 3$.

Clearly, by arguments given earlier,

$$P_O = P_R = \frac{1}{k} \tag{18}$$

in all three cases. However, P_T is dependent on i in these constructions for the factor array A. For $i = 1$, given any acceptable message the transmitter will be certain of the other message the receiver will accept, i.e., once he is informed of which message he is to use in the encoding rule constructed by the arbiter he knows the other message that the receiver will accept, since there is only one authenticating rule containing any given message. Consequently, he could send a message which the receiver would accept as authentic and which he could disavow with certainty that the arbiter would not hold him liable. Similarly, he knows that one of three possible messages must be the other acceptable message when $i = 3$, however, they occur uniquely in three equally likely authenticating rules, so that his probability of guessing which one the receiver will accept would be only 1/3 when $i = 3$. In the general case in which A is an i-net derived from an affine plane by choosing i spreads, the transmitter's probability of success will be

$$P_T = \frac{(k-1)}{i(k-1)} = \frac{1}{i}. \tag{19}$$

Since we want the code to be equitable, it must be the case that $i = k$ as is seen by equating the right-hand terms in (18) and (19). A necessary, but unfortunately not sufficient, condition for a code to be perfect is that it be equitable. We exhibit an example later which shows that being equitable does not imply perfection.

It is now easy to see why the procedure prescribed for the arbiter to use in the example of the previous section worked. The arbiter randomly chose a row from the Cartesian product of the lines in the missing spread for each of the factors, where each factor is itself a reduced affine plane. But a line from any spread intersects a line from any other in precisely one point, hence the net result of the arbiter's choice of a vector to use in forming the Shur product was to choose, with a uniform probability distribution, one message out of the $k = p^a$ messages used in each factor of each $\mathbf{a}_i$. The construction based on affine planes simply made it easy to accomplish this desired result.

The following example demonstrates that there do exist equitable $\mathbf{A}^2$-codes for which $\alpha > 1$ and $\lambda > 1$. The existence of these codes is at present only of interest for the sake of completeness, however, it appears that they may be useful in extending $\mathbf{A}^2$-codes to provide protection (to the transmitter and receiver) against the arbiter cheating. To construct one such factor array, A, we start with the balanced incomplete block design (BIBD) derived from $EG(3, 2)$, where the blocks are identified with the 14 planes in the geometry and the elements are identified with the points. A similar $\mathbf{A}^2$-code can be constructed starting with an arbitrary $EG(3, p^\beta)$ using the incidence properties of its planes. The parameters of this particular design are

$$\text{BIBD}(v, b, r, k, \lambda) = (8, 14, 7, 4, 3).$$

From the geometry of $EG(3, 2)$ we know that each plane (2-flat) has a parallel translate and that together each such pair covers the points of $EG(3, 2)$, i.e., the design is resolvable, and that furthermore any two planes that are not parallel intersect in a line (two-points) so that the design is also affine. The complete design, written in a form to show the resulutions, is

$$\begin{array}{llllllll}
\pi_1 & (0 & 1 & 2 & 3) & (4 & 5 & 6 & 7) \\
\pi_2 & (0 & 1 & 4 & 5) & (2 & 3 & 6 & 7) \\
\pi_3 & (0 & 2 & 4 & 6) & (1 & 3 & 5 & 7) \\
\pi_4 & (0 & 1 & 6 & 7) & (2 & 3 & 4 & 5) \\
\pi_5 & (0 & 2 & 5 & 7) & (1 & 3 & 4 & 6) \\
\pi_6 & (0 & 4 & 3 & 7) & (1 & 2 & 3 & 5) \\
\pi_8 & (0 & 3 & 5 & 6) & (1 & 2 & 4 & 7)
\end{array}$$

where the 3-tuple representations of the points in $EG(3, 2)$ have been replaced with the numerical value of the 3-tuples considered as binary numbers to give a more concise notation. To construct A we choose a subset of the parallel classes of the BIBD in such a way that four pairs of symbols (points) do not occur at all, and all others occur uniformly twice, i.e., so that $\lambda = 2$. To do this, select any pair and delete the three resolutions in which it occurs, i.e., choose any line in $EG(3, 2)$ and delete the three parallel classes on that line. For example, choosing the pair (0, 7) we delete classes π_4, π_5, and π_6 to get the array

$$\begin{array}{cc|cccccccc}
\pi_1 & a_1 & 1 & 1 & 1 & 1 & - & - & - & - \\
 & a_2 & - & - & - & - & 1 & 1 & 1 & 1 \\
\hline
\pi_2 & a_3 & 1 & 1 & - & - & 1 & 1 & - & - \\
 & a_4 & - & - & 1 & 1 & - & - & 1 & 1 \\
\hline
\pi_3 & a_5 & 1 & - & 1 & - & 1 & - & 1 & - \\
 & a_6 & - & 1 & - & 1 & - & 1 & - & 1 \\
\hline
\pi_3 & a_7 & 1 & - & - & 1 & - & 1 & 1 & - \\
 & a_8 & - & 1 & 1 & - & 1 & - & - & 1
\end{array},$$

where $k = 4$ and $\lambda = 2$. The pairs (0, 7), (1, 6), (2, 5), and (3, 4) do not occur at all in this array and all other pairs occur (uniformly) twice. Using this factor array as A, an opponent's probability of deceiving the receiver, where $\mathbf{A} = A^{\sigma}$ is the set of authenticating rules, is 1/2 for either impersonation and substitution. If the arbiter chooses only a single message from each factor in the authenticating rule chosen by the receiver to communicate each source state (using a uniform distribution), i.e., if $\alpha = 1$, the receiver's probability of being able to attribute successfully a fraudulent message to the transmitter will be

$$P_{\mathbf{R}} = 1/4$$

for both $\mathbf{R}_0$ and $\mathbf{R}_1$ deceptions. It is possible, however, to force $P_{\mathbf{R}}$ also to be 1/2 in this case. Let $\alpha = 2$ so that the arbiter is constrained to choose, again with a uniform probability distribution, one of the six possible pairs from among the four messages

in each factor a_i that the receiver selects. In this case, clearly,

$$P_R = 2/4 = 1/2.$$

Consider now the case of the transmitter disavowing a message. If $\alpha = 1$, then, for any choice (by the arbiter) of a message from a row $\mathbf{a}_i$, there will be a set of four other rows containing that message, any one of which could have been the one chosen by the receiver, and a total of six messages that could be paired with it. Only three of these messages occur in the authenticating rule chosen by the receiver. Hence,

$$P_T = 3/6 = 1/2.$$

For $\alpha = 2$, though, there will be only two rows containing the chosen pair, and, hence, only four possibly acceptable messages. Only two of these messages occur in the rule selected by the receiver. Therefore, in this case

$$P_T = 2/4 = 1/2$$

and the code is equitable. Consequently, for this choice of the factor array A, the resulting $\mathbf{A}^2$-code is equitable only for the case $\alpha = 2$. The design is not efficient in its use of the channel, however, since we already know that the bound of $P_D = 1/2$ can be satisfied with only four messages and four authenticating rules in the factor A, instead of eight of each as required in the array just shown. The point of this example was to show that there are arrays with $\alpha > 1$ and $\lambda > 1$ that satisfy (9), although we know of no maximally efficient designs for which this is the case.

This construction shows that being equitable does not imply perfection since in this case one more bit per factor is communicated than is used either to convey the state of the source or to confound one of the cheaters.

6. Postscript

The protocol described here to realize an $\mathbf{A}^2$-code from a Cartesian product array, $\mathbf{A} = A^\sigma$, in which the receiver chooses a row $\mathbf{a}$ of $\mathbf{A}$ that he secretly communicates to the arbiter, following which the arbiter chooses an encoding rule $\mathbf{e}$ from $\mathbf{a}$ which he in turn secretly communicates to the transmitter, is how we first conceived of $\mathbf{A}^2$-codes. There is another protocol—equivalent in the sense that the resulting code is the same and in which each of the participants ends up knowing precisely the same privileged pieces of information—that is logically more attractive. In this alternative protocol, the arbiter chooses from $\mathbf{A}$ the authentication code, i.e., set of acceptable messages, $\mathbf{a}$, which he secretly communicates to the receiver. He also chooses the encoding rule $\mathbf{e}$ from $\mathbf{a}$ exactly as in the existing protocol and secretly communicates $\mathbf{e}$ to the transmitter, etc. Nothing else is changed. The point is that in this protocol the only choices to be made in setting up the $\mathbf{A}^2$-code are made by the arbiter alone.

In the proof of Theorem 1 we made extensive use of the reduction of the authentication with arbitration scheme to simple authentication games: the opponent versus the receiver for I_0 and $\mathbf{S}_1$ deceits, the receiver versus the arbiter for $\mathbf{R}_i$

deceits, and the transmitter versus the receiver and the arbiter in the case of **T** deceits. The complication in the latter case was due to the encoding rule, **e**, being a joint result of a decision made by the receiver (the choice of **a**) and a choice (of **e** given **a**) by the arbiter. This is reflected in the definition of a successful **T** deceit, i.e., the transmitter is successful (in a **T** deceit) if and only if the message is accepted as authentic by the receiver but is not certified by the arbiter to be one that the transmitter would have used. The alternative formulation of $\mathbf{A}^2$-codes makes all of the choices involved in setting up the $\mathbf{A}^2$-code (i.e., strategies) be the responsibility of the arbiter, and consequently makes the arbiter the protagonist in all three games: the opponent versus the arbiter for $\mathbf{I}_0$ and $\mathbf{S}_1$ deceits, the receiver versus the arbiter for $\mathbf{R}_i$ deceits, and the transmitter versus the arbiter for **T** deceits. This is a logically neater and more symmetric protocol.

For either protocol there is a natural question of why—given α, k, and v—the authenticating rule, **a**, is not chosen with a uniform probability distribution from among all of the $\binom{v}{k}$ k-sets and the encoding rule, **e**, chosen similarly from among the $\binom{k}{\alpha}$ α-sets. Clearly, this would result in an $\mathbf{A}^2$-code having all of the properties of the codes described here. The answer is that the secret communications from the arbiter to the transmitter and to the receiver, or from the receiver to the arbiter depending on which one of the protocols is being used, must be made over secure and hence presumably expensive channels so that a premium is placed on minimizing the amount of information contained in these communications: this is the key distribution problem for the authentication channel. If all of the $\binom{v}{k}$ k-sets appeared in A and if the arbiter could use any one of the $\binom{k}{\alpha}$ α-sets in forming **e**, then $\sigma \log \binom{v}{k}$ bits would have to be exchanged securely between the arbiter and the receiver, which compares very unfavorably with the $\sigma \log(v)$ bits that need to be exchanged in the scheme proposed here. For $\alpha = 1$, the number of bits that need to be exchanged securely between the arbiter and the transmitter would be the same. This is only true for $\alpha = 1$, however; for $\alpha > 1$, the comparison is just as unfavorable as for the arbiter–receiver communication.

References

[1] T. Beth, D. Jungnickel, and H. Lenz, *Design Theory*, Bibliographisches Inst., Zurich, 1985.

[2] E. F. Brickell, A Few Results in Message Authentication (15th Southeastern Conference on Combinatorics, Graph Theory and Computing, Baton Rouge, LA, March 5–8, 1984), *Congressus Numerantium*, Vol. 43, 1984, pp. 141–154.

[3] E. F. Brickell and D. R. Stinson, Authentication Codes with Multiple Arbiters (Eurocrypt '88, Davos, Switzerland, May 25–27, 1988), in *Advances in Cryptology*, ed. by Christoph G. Günther, Springer-Verlag, Berlin, 1988, pp. 51–55.

[4] R. H. Bruck, Finite Nets, I: Numerical Invariants, *Canadian Journal of Mathematics*, Vol. 3, 1951, pp. 94–107.

[5] R. H. Bruck, Finite Nets, II: Uniqueness and Embedding, *Pacific Journal of Mathematics*, Vol. 13, 1963, pp. 421–457.

[6] M. De Soete, Some Constructions for Authentication—Secrecy Codes (Eurocrypt '88, Davos, Switzerland, May 25–27, 1988), in *Advances in Cryptology*, ed. by Christoph G. Günther, Springer-Verlag, Berlin, 1988, pp. 57–75.

[7] T. El Gamal, A Public Key Cryptosystem and a Signature Scheme Based on Discrete Logarithms, *IEEE Transactions on Information Theory*, Vol. 31, No. 4, 1985, pp. 469–472.

[8] E. N. Gilbert, F. J. MacWilliams, and N. J. A. Sloane, Codes Which Detect Deception, *The Bell System Technical Journal*, Vol. 53, No. 3, March 1974, pp. 405–424.

[9] S. Goldwasser, S. Micali, and R. L. Rivest, A Digital Signature Scheme Secure Against Adaptive Chosen-Message Attacks, *SIAM Journal on Computing*, Vol. 17, No. 2, 1988, pp. 281–308.

[10] M. Hall, Jr., *Combinatorial Theory*, second edition, Wiley, New York, 1986.

[11] C. L. Henderson and A. M. Fine, Motion, Intrusion and Tamper Detection for Surveillance and Containment, Report SAND79-0792, Sandia National Laboratories, March 1980; also published by the International Safeguards Project Office for the International Atomic Energy Agency (IAEA) as ISPO Report No. 91, 1980.

[12] J. L. Massey, Cryptography—A Selective Survey (International Tirrenia Workshop on Digital Communications, Tirrenia, Italy, Sept. 1–6, 1985), *Alta Frequenza*, Vol. LV, No. 1, 1986, pp. 4–11.

[13] P. D. Merrillat, Secure Stand-alone Positive Personnel Identity Verification System (SSA-PPIV), Technical Report SAND79-0070, Sandia National Laboratories, March 1979.

[14] J.-J. Quisquater and J.-P. Delescaille, How Easy Is Collision Search? Application to DES (Eurocrypt '89, Houthalen, Belgium, April 11–13, 1989, updated version Crypto '89, Santa Barbara, CA, August 20–24, 1989), in *Advances in Cryptology*, ed. by G. Brassard, Springer-Verlag, Berlin, 1990, pp. 419–424.

[15] M. O. Rabin, Digitized Signatures and Public-key Functions as Intractable as Factorization, Technical Report LCS/TR-212, M.I.T. Laboratories for Computer Science, 1979.

[16] R. Rivest, A. Shamir, and L. Adleman, A Method for Obtaining Digital Signatures and Public Key Cryptosystems, *Communications of the ACM*, Vol. 21, No. 2, 1978, pp. 120–126.

[17] P. Schoebi, Perfect Authentication Systems for Data Sources with Arbitrary Statistics, Presented at Eurocrypt '86, Linköping, Sweden, May 20–22, 1986.

[18] G. J. Simmons, A Game Theory Model of Digital Message Authentication (11th Annual Conference on Numerical Mathematics and Computing, Winnipeg, Oct. 1–3, 1981), *Congressus Numerantium*, Vol. 34, 1982, pp. 413–424.

[19] G. J. Simmons, A System for Verifying User Identity and Authorization at the Point-of-Sale or Access, *Cryptologia*, Vol. 8, No. 1, 1984, pp. 1–21.

[20] G. J. Simmons, Message Authentication: A Game on Hypergraphs (15th Southeastern Conference on Combinatorics, Graph Theory and Computing, Baton Rouge, LA March 5–8, 1984), *Congressus Numerantium*, Vol. 45 1984, pp. 161–192.

[21] G. J. Simmons, Authentication Theory/Coding Theory (Crypto '84, Santa Barbara, CA, August 19–22, 1984), in *Advances in Cryptology*, ed. by R. Blakley, Springer-Verlag, Berlin, 1984, pp. 411–431.

[22] G. J. Simmons, Authentication Codes that Permit Arbitration (18th Southeastern Conference on Combinatorics, Graph Theory and Computing, Boca Raton, FL, Feb. 23–27, 1987), *Congressus Numerantium*, Vol. 59, 1987, pp. 275–290.

[23] G. J. Simmons, Message Authentication with Arbitration of Transmitter/Receiver Disputes (Eurocrypt '87, Amsterdam, April, 13–15, 1987), in *Advances in Cryptology*, ed. by D. Chaum and W. L. Price, Springer-Verlag, Berlin, 1988, pp. 151–165.

[24] G. J. Simmons, A Natural Taxonomy for Digital Information Authentication Schemes (Crypto '87, Santa Barbara, CA, Aug. 16–20, 1987), in *Advances in Cryptology*, ed. by Carl Pomerance, Springer-Verlag, Berlin, 1988, pp. 269–288.

[25] G. J. Simmons, A Protocol To Provide Verifiable Proof of Identity and Unforgeable Certified Receipts, *IEEE Journal of Selected Areas in Communications* (Special Issue on Secure Communications), Vol. 7, No. 4, 1989, pp. 435–447.

[26] D. R. Stinson, Some Constructions and Bounds for Authentication Codes (Crypto '86, Santa Barbara, CA. Aug. 12–15, 1986), in *Advances in Cryptology*, ed. by A. M. Odlyzko, Springer-Verlag, Berlin, 1987, pp. 418–425; also in *Journal of Cryptology*, Vol. 1, No. 1, 1988, pp. 37–51.

[27] D. R. Stinson, A Construction for Authentication Secrecy Codes from Certain Combinatorial Designs, (Crypto '87, Santa Barbara, CA, Aug. 16–20, 1985), in *Advances in Cryptology*, ed. by Carl Pomerance, Springer-Verlag, Berlin, 1988, pp. 355–366; also in *Journal of Cryptology*, Vol. 1, No. 2, 1988, pp. 119–127.

[28] H. C. Williams, A Modification of the RSA Public-Key Encryption Procedure, *IEEE Transactions on Information Theory*, Vol. 26, No. 6, 1980, pp. 726–729.