A Natural Taxonomy for Digital Information Authentication Schemes*

Gustavus J. Simmons
Sandia National Laboratories
Albuquerque, New Mexico 87185

There are two objectives that prompt the authentication of information; one is to verify that the information was, in all probability, actually originated by the purported originator, i.e., source identification, the other is to verify the integrity of the information, i.e., to establish that even if the message was originated by the authorized source, that it hasn't been subsequently altered, repeated, delayed, etc. These two objectives are normally treated in the theory of authentication as though they are inseparable, and will also be treated in that way here, although recent results by Chaum [1] demonstrating message integrity with source anonymity and by Fiat and Shamir [2], by Goldreich, Micali and Wigderson [3], and by others demonstrating verification of source identity with no additional information exchange show that the functions can in some instances be separated. The relevance of this comment to the subject matter of this paper is that it suggests that there may be a fourth independent coordinate in information authentication besides the three that will be discussed here. In spite of considerable effort, we have been unable to produce a convincing argument for or against this being the case, so we only mention the possibility for completeness.

In deference to the origins of the problem of authentication in a communications context, we shall refer to the authenticated information as the message and to the originator (of a message) as the transmitter. The message, devoid of any meaningful physical embodiment, is presented for authentication by a means that we call the authentication channel. In the simplest possible authentication scheme, the party receiving the message (the receiver) is also the one wishing to verify its authenticity; although, as we shall see, 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 can equally well be data in a computer file or resident software in a computer; it can be quite literally a fingerprint in the application of the authentication channel to the verification of the identity of an individual [4,5] or figuratively a "fingerprint" in the verification of the identity

* This work performed at Sandia National Laboratories supported by the U. S. Dept. of Energy under contract no. DE-AC04-76DP00789.

C. Pomerance (Ed.): Advances in Cryptology - CRYPTO '87, LNCS 293, pp. 269-288, 1988.

of a physical object such as a document or a tamper sensing container [6]. In the broadest sense, authentication is concerned with establishing the integrity of information strictly on the basis of the internal structure of the information itself, irrespective of the source of the information.

In a series of papers [7,8,9,10], the present author has developed a mathematical model for the authentication channel in which sharp bounds are derived for the probability, P_d, that an i.e., either a message
sent by an unauthorized transmitter or else a message from an authorized transmitter that has been intercepted and either modified or replaced by a substitute message, to be accepted by the receiver as authentic. One of these bounds, first derived by Gilbert, MacWilliams and Sloan [11] in a slightly different setting than described above is;

$$P_d \geq \frac{1}{\sqrt{|\mathcal{E}|}} \tag{1}$$

where $|\mathcal{E}|$ is the total number of rules available to the transmitter/receiver for encoding states of the source into messages*. In 1985, we reported [10] a paradoxical situation in which the bound in (1) appeared to be violated. The explanation of this paradox is the source of the first dichotomy on which the taxonomy of authentication schemes is based.

Consider the following simple example: a transmitter, the Tx, wishes to communicate one of two equally likely instructions to a receiver, the Rx, say, to buy or sell a particular (agreed upon in advance) stock. The point being that precisely one bit of information is to be communicated from the Tx to the Rx, any other one-bit source, i.e., a toss of a fair coin, would serve equally well. The communication must take place over a publicly exposed communications channel on which a third party, known as the eavesdropper (or opponent), is listening. This channel is often described as a party line telephone when only secrecy is of concern, although a party line is not a completely adequate model for authentication as we will see later.

In the case of secrecy, i.e., the classical objective of encryption, it is assumed to be vital to the Tx's and Rx's interests that the eavesdropper(s) not know which instruction the Tx is sending to the Rx. The following simple protocol, known to cryptographers as Vernam or one-time key encryption, insures that the eavesdrop-

* Ideally we would call the information to be authenticated "messages" as is the practice in communications theory. However, if we adopt this convention we are forced to introduce terminology to designate the collection of sequences that can be sent through the channel. We would either have to coin a new word to designate the particular sequence of symbols sent to convey and authenticate a message -- none of which seem very natural -- or else use the cumbersome term "authenticated message". The term "authenticator", usually used in the sense of an authentication code word appended to a message, has too restricted a connotation for the general case. We have opted instead to use the term "message" to designate what is actually transmitted and to tolerate the rather artificial device that the information conveyed by a message is the state of a hypothetical source.

per's a priori and a posteriori probabilities of determining the Tx's instruction to the Rx are the same. In other words, eavesdropping on the communication channel doesn't help the opponent to deceive the receiver. In order to foil the eavesdroppers, the Tx and Rx agree in advance as to whether the Tx will speak truthfully when he says buy or sell, or whether he will lie in what he says. Since the information content of an instruction in this example is one bit, they must introduce at least one bit of equivocation (to the eavesdropper) about the Tx's actions. They do this by flipping a fair coin and using the outcome with the following protocol to decide on the Tx's course of action, i.e., whether he will lie or speak truthfully in his instructions to the Rx.

		Cipher	
		Buy	Sell
Key	H	Buy	Sell
	T	Sell	Buy

← Instruction

If heads comes up, the Tx will say "Buy" when he wants the Rx to buy and "Sell" when he wants the Rx to sell. If tails comes up, however, he will say "Sell" when he wants the Rx to buy, and so forth. It should be clear that by using the protocol in this way the eavesdropper will know no more about the actual (encrypted) instruction the Tx sent to the Rx as a result of listening in on their telephone conversation than he would have had he not listened at all. Such a cryptosystem was defined by Shannon to be perfect -- in the obvious meaning of the term that the a priori and a posteriori (after observing a legitimate cipher sent by the Tx) probabilities of the eavesdropper being able to determine the instruction are the same.

The cryptographic key in this simple example is the knowledge (shared by the Tx and Rx) of whether the Tx is telling the truth or not: encryption is the act by the Tx of either speaking truthfully or lying as determined by the key, the cipher is what the Tx says while decryption is the interpretation by the Rx of what the Tx actually meant, not necessarily what he said. It should be obvious that the Tx and Rx cannot reuse the key in this example to encrypt a second instruction since the eavesdropper could determine the key that was used (after the fact) by comparing the action taken by the Rx with the observed cipher (instruction ?). In order for this protocol to be secure for repeated communications, the Tx and Rx must secretly exchange in advance of any communication as much information in the form of keys (coin flips) as they later wish to communicate as encrypted instructions. This requirement for advanace secret key exchange and the associated key protection problem is a serious practical limitation to the usefulness of perfect encryption schemes, which, incidentally, are the only schemes whose cryptosecurity are presently mathematically demonstrable. The relevance of this example to the present discussion of authentication is that the security provided by this secrecy protocol is provable, i.e., it is independent of the computing power an opponent may bring to bear on

breaking a cipher. Although it is not essential to the purposes of the present example, it should be pointed out that there are similar provably secure encryption protocols for arbitrary security requirements.

This example can be extended to illustrate the authentication of the one bit of information, i.e., of providing a means for the Rx to verify (with some calculable confidence) that a message actually came from the legitimate (authorized) Tx and not from someone impersonating the Tx and that it has not been altered subsequent to the legitimate Tx having sent it. In the secrecy protocol just described, if the opponent were in a position to not only listen to communications from the Tx but also to either send fraudulent ciphers (pretending to be the Tx) or else to intercept legitimate ciphers sent by the Tx and substitute others of his own devising then he could be certain of deceiving the Rx irrespective of which strategy he chooses. If he intercepts the Tx's communication (cipher), he could, even though he cannot interpret the cipher, cause the Rx to act contrary to the Tx's intention by simply substituting the other cipher for the one actually sent by the Tx. Similarly, although he would not know which action the receiver would take since he does not know the key chosen by the Tx and Rx, he could send either cipher, "Buy" or "Sell", with the assurance that it would be accepted and acted on by the Rx. In either event, the opponent would be certain of deceiving the Rx to act in a way not requested by the Tx.

To protect against this sort of deception by outsiders, there is essentially only one strategy available to the Tx and Rx. They must enlarge the set of messages that can be sent through the channel so that for any particular choice of an encoding rule (corresponding to a choice of a key for the secrecy channel) there will be some messages that will be acceptable to the Rx, i.e., that the Tx might send to the Rx according to the encoding rule (protocol), while others would be rejected as unauthentic since the Tx would not have used them (under the chosen encoding rule). In other words, the opponent must be uncertain of which messages will be acceptable to the receiver in all cases. Message authentication is critically dependent on this uncertainty, and on how it is distributed over encoding rules, messages and source states. Ideally, in analogy to perfect encryption schemes, this should be done in such a way that an opponent has no better chance of deceiving the Rx if he waits and observes a legitimate message than he would have had, had no observation been made. Again, in the smallest possible example, i.e., of a one-bit source, there are two equally likely instructions, say buy and sell as before. Instead of two messages however, we shall now use four, which requires that two bits actually be communicated. These two bits, if used optimally will inform the Rx of the Tx's intention (one bit) and provide precisely one bit of authentication; i.e., the opponent's probability of deceiving the Rx will be 1/2 irrespective of whether he chooses to impersonate the Tx or to wait and observe a legitimate message and then substitute some other message in its stead.

Probably the most commonly encountered authentication scheme in practice is one that uses an authenticator appended to the Tx's intended communication, say Hi or Lo

appended to the instruction to buy or sell for this example. The four messages in this case would then be Buy-Hi, Buy-Lo, Sell-Hi and Sell-Lo where the first part of the message is the (unencrypted in this example) instruction and the second part is the appended authenticator. In the example there are also four encoding rules (corresponding to keys in the encryption example), a particular one of which is chosen with uniform probability distribution by the Tx and Rx flipping a fair coin twice in advance of their needing to authenticate a message (and in secret from the opponent, denoted by the labels HH, HT, TH and TT. The specific authentication protocol we will discuss, out of several possible protocols satisfying the conditions of this example, is the Cartesian product construction

$$\begin{matrix} H\begin{pmatrix} Buy & - \end{pmatrix} \\ T\begin{pmatrix} - & Buy \end{pmatrix} \end{matrix} \otimes \begin{matrix} H\begin{pmatrix} Sell & - \end{pmatrix} \\ T\begin{pmatrix} - & Sell \end{pmatrix} \end{matrix} =$$

	Buy-Hi	Buy-Lo	Sell-Hi	Sell-Lo
HH	Buy		Sell	
TT	Buy			Sell
HT		Buy	Sell	
TH		Buy		Sell

If the opponent knowing the protocol shown above but not the particular encoding rule chosen by the Tx and Rx, attempts to impersonate the Tx and send a message to the Rx, since there are four equally likely encoding rules, for each of which there are only two acceptable (to the Rx) messages, there is clearly only a 50-50 chance that the message the opponent chooses will be one of the two acceptable ones for the particular encoding rule being used by the Tx and Rx. If, on the other hand, the opponent waits to observe a message sent by the Tx, his uncertainty about the encoding rule they are using will have shrunk from one in four equally likely possibilities to one of two. However, there is a single, but different, acceptable substitute message in each case depending on which rule is being used by the Tx and Rx. For example, if the observed message is Buy-Hi, then the encoding rule must be one of the pair labeled HH and HT. In the first case Sell-Hi would be accepted by the receiver and Sell-Lo would be rejected as unauthentic, while the reverse would be true if the chosen encoding rule was the one labeled HT. This example illustrates a two-bit authentication scheme for which equality holds in (1) that communicates one bit of information and provides one bit of authentication capability at the expense of transmitting two bits of information in each message. The important point to this example is that the confidence the transmitter and receiver can have in the authentication of their communication is provable, i.e., independent of the computational capabilities of an opponent. Brickell [12], Stinson [13,14], Simmons [7,8,9,10] and others have devised various provably secure authentication schemes or codes. The bound in (1) applies to all of these codes, many of which are also perfect in the sense that equality holds. These codes make it possible to provide arbitrarily high levels of confidence in the authenticity of messages, at the expense of a very large usage of key-like secret information defining the selection of encoding rules to be used.

For many real-world applications, the authentication "key" distribution problem described above is avoided by using cryptographic concealment of the sets of acceptable and unacceptable messages. Considering again the one-bit source example -- the outcome of a fair-coin toss in this case -- used earlier, the transmitter and receiver could use standard encryption techniques to generate the sets of messages which the transmitter will send and the receiver will accept. With no loss of generality, it is assumed that it is public knowledge that the transmitter and receiver have agreed that the 64-bit binary sequence 11...1 will denote the outcome "heads" and the string 01...1 will denote "tails". In other words, the redundant information used to authenticate a message is the suffix of 63 1's and only the left-most bit in a sequence conveys the outcome of the coin toss. To protect themselves from deception by the opponent the transmitter/receiver arrange (in advance) to encrypt whichever of these sequences the coin toss indicates using the data encryption standard (DES) cryptoalgorithm and a secret (known only to them) DES key, which, as is well known, consists of 56 bits of equivocation to an outsider, the wiretapper. Each of the 2^{56} possible choices of a DES key corresponds in this scheme to a choice of an authentication encoding rule. Consequently, $|\mathscr{E}| = 2^{56}$, and the bound (1) says that even if the transmitter/receiver choose among the 2^{56} encoding rules optimally, they cannot limit the opponent's probability of successfully deceiving the receiver into accepting an unauthentic message to less than

$$P_d \geq \frac{1}{\sqrt{|\mathscr{E}|}} = \frac{1}{2^{28}} \approx 3.7 \times 10^{-9} \tag{2}$$

or roughly four parts in a billion.

In practice, the opponent's chances of success are dramatically less than (2) would suggest. There are 2^{64} possible ciphers (messages), only two of which are acceptable for any particular choice of a key (authentication encoding rule). Therefore, if the opponent merely selects a cipher at random and attempts to impersonate the transmitter, his probability of success is 2^{-63} or approximately one chance in 10^{19}. The question is, can he do better. As far as impersonating the transmitter is concerned, the answer is essentially no, even if he has unlimited computing power. For each choice of an encoding rule, there are two (out of 2^{64}) ciphers that will be acceptable as authentic. Assuming that the mapping of the sequences 11...1 and 01...1 into 64-bit cipher sequences under DES keys is a random process, this says that the total expected number of acceptable ciphers (over all 2^{56} keys) is $\approx 2^{56.9888}$, i.e., ϵ close to 2^{57}. Even if the opponent could feasibly carry out the enormous amount of computation that would be required to permit him to restrict himself to choosing a cipher from among this collection, his chances of having a fraudulent message be accepted by the receiver would still only be $\approx 2^{-56}$ or roughly one chance in 10^{17} which is what we meant when we said that the answer was essentially no since 10^{-17} isn't much different than 10^{-19} while both differ enormously from the bound, (2), of $\approx 10^{-9}$. The opponent could not do better, nor worse, (in attempting

to impersonate the transmitter) even if he possessed infinite computing power than choose a cipher randomly, with a probability distribution weighted to reflect the number of times each cipher occurs, from among the $\approx 2^{57}$ potentially acceptable ciphers.

However, the channel bound in (1) applies to all authentication schemes, hence the apparent contradiction must arise in connection with the opponent's substitution strategy. If the opponent waits to observe a legitimate message (cipher), can the information acquired by virtue of this observation be put to practical use to improve his chances of deceiving the receiver? Even if he doesn't know the state of the source, he knows that the cipher is the result of encrypting one of the two 64-bit sequences 111...1 or 011...1 with one of the 2^{56} DES keys. He also knows that with a probability of essentially one (≈ 0.996), there is only one key that maps the observed cipher into either of these two sequences, hence, he is faced with a classical "meet in the middle" cryptanalysis of DES. Clearly if he succeeds in identifying the DES key, i.e., the encoding rule being employed by the transmitter receiver, he can encrypt the other binary string and be certain of having it be accepted, and hence be certain of deceiving the receiver. The point, though, is that in order for him to make use of his observation of a message he must be able to determine the DES key the transmitter and receiver(s) are using, i.e., he must be able to cryptanalyze DES. If he can do this, the expected probability of deceiving the receiver is ϵ close to one, the small deviation being attributable to the exceedingly small chance that two (or more) DES keys might have encoded source states into the same message (cipher). Thus, we have the paradoxical result that the practical system is some eight or nine orders of magnitude more secure than the theoretical limit simply because it is computationally infeasible for the opponent to carry out in practice what he should be able to do in principle. In this respect, practical message authentication is closely akin to practical cryptography where security is equated to the computational infeasibility of inverting from arbitrarily much known matching cipher text and plaintext pairs to solve for the unknown key, even though in principle there is more than enough information available to insure a unique solution.

The example of the preceding paragraphs illustrates very clearly the first essential dichotomy in authentication schemes, namely the division according to whether the security of the authentication is provable, i.e., independent of the computing power and time the opponent may bring to bear, or else dependent on the infeasibility of his being able to carry out in practice a computation that in principle he could do:

provably secure -- computationally secure

To demonstrate the essential nature of the next dichotomy in authentication schemes, we need to briefly examine the two most widely used schemes at present. Authentication has traditionally been achieved by way of encryption using single-key

cryptoalgorithms since, until recently, this was the only type of cryptography available. In a common U. S. military authentication protocol, for example, both the transmitter and receiver are provided with a matching pair of sealed authenticators distributed with the same physical security with which the cryptographic keys are handled; actually a random sequence of symbols produced and distributed by the National Security Agency. The sealed packets are constructed so as to provide a positive indication (tattle-tale) if they are opened. Each communicant is responsible for the protection of his sealed authenticator and is administratively constrained from opening it until it is to be used. To authenticate a message, the transmitter opens his sealed packet, appends the enclosed authentication suffix to the message and then either block encrypts the resulting extended message or else encrypts it with cipher or text feedback so that the effect of the appended authenticator is spread throughout the resulting cipher under the control of the secret key. This cipher is then transmitted as the authenticated message. The receiver, upon receiving and decrypting the cipher, opens his matching sealed authenticator and accepts the message as genuine if the cipher decrypted to a string of symbols with an authenticating suffix matching his authenticator, and otherwise rejects it as unauthentic.

Because of the sensitivity of the authenticators, i.e., anyone having access to one could authenticate a fraudulent message of his own choice, they must generally be handled under two-man control both in distribution and in the field prior to their use which greatly complicates their distribution and control, and more importantly limits the physical environments in which they can be used. Since the key in a single-key cryptoalgorithm must also be handled in the same way, the sealed authenticators have only marginally affected the physical security requirements, and hence have generally been acceptable for military and diplomatic communications. In addition, and even more significant for the present discussion, the cipher that is the authenticating message must be completely inscrutable to an outsider, otherwise it would be possible to strip an authenticator off an authentic message and attach it to a fraudulent one. The point is that while this classic military authentication scheme does provide secure authentication of information, it does so at the expense of requiring complete secrecy for the information being authenticated.

On the other hand, the authentication of electronic funds transfers in the Federal Reserve System does not require nor result in secrecy for the information being authenticated. By directive of the Secretary of the Treasury [15], all such transfers must be authenticated using a procedure that de facto depends on the Data Encryption Standard (DES) single-key cryptographic algorithm. The protocol includes precise format requirements, etc., however the essential feature for our purposes is that an authenticator is generated using a DES mode of operation known as block chaining. The information to be authenticated is first broken into blocks of 64-bits each. The first block is added modulo two (exclusive or) to a 64-bit initial vector (IV), which is changed daily and kept secret, and the sum encrypted using a secret

DES key (known to both the transmitter and the receiver). The resulting 64-bit cipher is then exclusive or'ed with the second block of text and the result encrypted to give a second 64-bit cipher, etc. This procedure is iterated until all blocks of the text have been processed. The final 64-bit cipher is clearly a function of the secret key, the initial vector, and of every bit of the text, irrespective of its length. This cipher is appended as an authenticator to the information being authenticated to form an extended message which is normally transmitted over an open communications channel, although it may be superencrypted if privacy is desired. However, this operation is independent of the authentication function. The authenticator can be easily verified by anyone in possession of the secret key and the initial vector by simply repeating the procedure used by the transmitter to generate it in the first place. An outsider, however, cannot generate an acceptable authenticator to accompany a fraudulent message, nor can he separate an authenticator from a legitimate message to use with an altered or forged message since the probability of it being acceptable in either case is the same as his chance of "guessing" an acceptable authenticator, i.e., one in 2^{64}. In this authentication scheme, which is a classic example of an appended authenticator, the authenticator is a complex function of the information that it authenticates, as well as the secret key and initial vector.

The important point which this example illustrates is that unlike the classic military authentication scheme where secrecy was an essential part of being able to authenticate information, the EFT authentication scheme does not require that the information being authenticated be kept secret. There is no particular reason for it not being secret, and as was indicated there are instances in which the extended message may be superencrypted (independent of authentication) to provide privacy but the authentication procedure does not itself conceal the information. Because of the intrinsic nature of single-key cryptography, however, the appended authenticator in the extended message is necessarily inscrutable to an outsider not in possession of the key, since anyone possessing the secret key and initial vector is, in addition to being able to verify the appended authenticator to a legitimate message, also able to authenticate an arbitrary fraudulent message.

In the early 70's, Sandia encountered the problem of authenticating data from seismic stations that had been designed to verify compliance (by the Russians) with a proposed comprehensive nuclear weapons test ban treaty [16]. Secrecy was not possible in this application since the Russians had to be able to "see" the information being communicated in order for the scheme to be acceptable to them, otherwise the U. S. could have conceivably transmitted information other than what had been agreed to by treaty. On the other hand, in order for the system to be acceptable to the Americans, it had to be true that the Russians, even though they could examine the authenticated message and verify the information it contained, not be able to utter a fraudulent message which the U. S. would accept as authentic. Apparently this application was the first in which authentication without secrecy was required.

Recall that the first discussion of two-key (read also public-key) cryptography in the open literature occurred several years later so that the only authentication schemes available for a system that was to be shared with the Russians in 1972 had to depend on conventional single-key cryptographic techniques, applied so as to approximate the desired end of authentication (to the monitor) without secrecy (to the host) in a scheme very similar to the EFT scheme described above. The compromise solution, found by Simmons, Stewart and Stokes in 1972 [17], was to form an authenticator that was much shorter than the message, where the authenticator was made to be a function of the entire message by repeated encryptions of blocks of text operated on by intermediate ciphers. The authenticator was then block encrypted and appended to the unencrypted message. This authentication protocol is currently referred to as a message authentication code (MAC) [18], and has a much cleaner implementation, for example, in the EFT authentication protocol. This solved the problem of making it possible for the host to monitor the seismic data in real time as it was transmitted, however, the encrypted authenticator was still inscrutable until he was later given the key with which it had been encrypted. Ironically, the host and monitor each trusted the resulting system to the same level of confidence for the same reason. The monitor trusts the authentication since in order to create a forgery the host would have to invert from a known plaintext/cipher pair, i.e., break the cryptosystem by cryptanalysis, to find the key used by the monitor. On the other hand, the host is satisfied that the monitor didn't conceal information in the preceding transmission if the key he is given generates the authenticator that was transmitted since the monitor would have had to solve for the (unique?) key relating the plaintext and a desired bogus authenticator that concealed a hidden message if he wished to cheat; i.e., the monitor would have to solve precisely the same hard problem on which he bases his confidence in the authenticator.

To shorten the periods of implicit trust required of the host, keys can be generated sequentially by the same cryptoalgorithm used to encrypt the authenticator so that for all intents there is an unlimited number of session keys available. This makes it feasible to process shorter blocks of data using a unique session key for each block, with a flow of session keys being made available to the host after essentially only the delay of a two-way satellite relay link. In the limit, with block size and the two-way delay, such a scheme is asymptotic to a true message authentication without secrecy system, although there is a period in each iteration during which the host is at greater risk of being cheated than is the monitor.

The discussion of the preceding paragraphs defines the second essential dichotomy in authentication schemes, namely whether the authentication is carried out with or without secrecy. Using single-key cryptographic techniques, it does not appear to be possible to completely realize the goal of having no uncertainty (secrecy) in the authenticated message, although we have described a procedure that gets asymptotically close to this objective. As we shall see in the next section, by using two-key

cryptographic techniques it is possible to realize authentication without secrecy, with no compromise required. The second dichotomy to authentication though is:

with secrecy -- without secrecy

In all of the authentication schemes discussed thus far, since the transmitter and receiver must both know the same secret (from the opponent) information (either the key in a simple key cryptographic algorithm, or a sealed authenticator and a cryptographic key or an initial vector and a cryptographic key, etc.) they can each do anything the other can do. In particular, because of this duality, the receiver cannot "prove" to a third party that a message he claims to have received from the transmitter was indeed sent by the transmitter, since he (the receiver) has the capability to utter an indetectable forgery, i.e., the transmitter can disavow a message that he actually sent. Similarly, the receiver can claim to have received a message when none was sent, i.e., to falsely attribute a message to the transmitter, who cannot prove that he didn't send the message since he could have. In the classic military authentication scheme this is an acceptable situation, since a superior commander doesn't worry that a subordinate will attribute an order to him that he didn't issue and the subordinate doesn't worry that his superior will disavow an order that he did send. There is, in fact, some rudimentary protection against this sort of cheating provided by the sealed authenticators the military uses since if either party can produce his unopened authenticator, it is prima facie evidence that he doesn't know its contents and hence could not have authenticated a message using its contents. In many situations, and in almost all commercial and business applications, the primary concern is with insider cheating, i.e., the person withdrawing cash from an ATM may not be the account holder or the amount shown on a properly signed and valid check may be altered to a larger figure, etc. In a more general model of message authentication, there are four participants instead of three. As before, there is a transmitter who observes an information source and wishes to communicate these observations to a remotely located receiver over a publicly exposed, noiseless, communications channel; a receiver who wishes to not only learn the state of the source (as observed by the transmitter) but also to assure himself that the communications (messages) he accepts actually were sent by the transmitter and that no alterations have been made to them subsequent to the transmitter having sent them and an who opponent wishes to deceive the receiver into accepting a message that will misinform him as to the state of the source. In addition, there is a fourth party, the arbiter. The arbiter's sole function is to certify on demand whether a particular message presented to him is authentic or not, i.e., whether it is a message that the transmitter could have sent under the established protocol. He can never say that the transmitter did send the message, although the probability that it could have come from a source other than the authorized transmitter can be made as small as one likes, only that he could have under the established protocol.

Since we wish to use the problem of authenticating seismic data to verify compliance with a comprehensive nuclear weapons test ban treaty to illustrate the third dichotomy, we return briefly to the problem of achieving true message authentication without secrecy. Two-key (nee public key) cryptography provides an immediate solution to this problem since the essential property of two-key cryptography is the separation of the secrecy channel from the authentication channel, which are inextricably linked in single-key cryptosystems. In two-key cryptography, the encrypt and decrypt keys are not only different, but it is also computationally infeasible to determine at least one of the keys from a knowledge of the other key and of arbitrarily many matched plaintext message/cipher pairs. If the receiver (decrypt) key cannot be deduced from a knowledge of the transmitter (encrypt) key, then the transmitter key may be publicly exposed, so long as the receiver key is kept secret, without jeopardizing the transmitter's ability to communicate in secret to the receiver, although the receiver cannot authenticate the source of the communication, i.e., cannot be sure of the origin of the ciphers. This is the secrecy channel. Conversely, if the transmitter's encrypt key cannot be recovered from a knowledge of the receiver's decrypt key, etc., then, although secrecy is impossible, the receiver can be confident that the communication originated with the purported transmitter and that the message has not been altered in transit if the transmitter can be unconditionally trusted to keep the encrypt key secret. This is the authentication channel.

The obvious solution to the authentication without secrecy problem exploits the authentication channel. The U. S. would install a two-key cryptographic system along with the seismic sensor package in the borehole with a secret (known only to the U. S.) encrypt key that would be volatilized if the package was tampered with. The decrypt key would be shared with the Russians (and perhaps with third parties who need to be able to authenticate transmissions). The messages are the ciphers obtained by encrypting the seismic data along with agreed upon identifiers -- station ID number, date, clock, message number, etc., -- which are required, not only for their obvious utility, but also to provide the redundant information needed by the U. S. to authenticate the messages. The Russians could decrypt the ciphers in real time, perhaps even delaying the transmission in a data buffer for the time required to decrypt, to satisfy themselves that nothing other than the agreed upon seismic data and prearranged formatting information were present. Thus no part of the transmission would have to be kept secret from the Russians. Similarly, the U. S. would decrypt the cipher upon receipt and accept the transmission as authentic if and only if the expected redundant formatting information was present. This scheme depends only on the availability of an authentication channel, separate from the secrecy channel, and hence is not dependent on any particular two-key cryptoalgorithm.

Unfortunately, although the system just described allows the monitor to authenticate messages to whatever level of confidence he requires while at the same time permitting the host to reassure himself that no unauthorized information is concealed,

it does not permit arbitration of disputes between the host and the monitor. If unilateral response by the monitor, such as abrogation of a treaty or resumption of atmospheric testing of nuclear weapons as the U. S. did in 1962 in response to the Soviets 1961 violation of the Joint Understanding of a moratorium on such tests, is the only action to result from a detection by the monitor of a violation of the agreement, the compromise system just described suffices since the monitor can be unilaterally convinced of the authenticity of the seismic information that indicates a violation of the treaty. If, however, the action to be taken by the monitor in the event that a violation is detected involves convincing third parties or arbiters, such as the United Nations, NATO, etc., then it must be impossible for the monitor to forge messages. Otherwise, the host could disavow an incriminating message as being a forgery fabricated by the monitor (i.e., disinformation in the current Washington terminology), an assertion which the monitor can not disprove since he has the known capability to encrypt messages and hence to create undetectable forgeries. The problem is that an acceptable (to the monitor) authentication scheme for this application must also include a capability to logically prove the authenticity of information to impartial third parties.

Various cooperative schemes were considered in which each of the three legitimate participants, the transmitter, the receiver and the arbiter(s), contributed to the key in a two-key algorithm in such a way that the resulting combined key was totally uncertain to each of them; for example, the exclusive or of three independent keys. Practical difficulties having to do with trust in equipment as opposed to the logical soundness of the protocol, finally forced the concatenation of three (or more if there is more than one arbiter) separate and independent two-key authentication channels. Each user supplies authentication equipment of his own construction operating with an encryption key that only he knows. His equipment operates on data and cipher streams from other equipments and communicates cipher streams to the other equipments and to the instrumentation cab at the top of the borehole. The decryption keys are shared by all parties. While it is true that each party can perform any operations that his -- supposedly secret -- downhole equipment can, this doesn't make it possible for him to utter an acceptable forgery since the other cryptosystems are inscrutable to him since he does not know the (secret) encryption keys they are using. Furthermore, any party by publicizing the secret information he is supposed to protect could only make it possible for the other parties to duplicate the actions of two out of the three or more encryption systems. This concatenated encryption system renders it impossible for the host to disavow incriminating messages by unilaterally compromising his key. From the monitor's standpoint, even if the host and the arbiter collude to deceive him, he will still be able to establish, to his satisfaction, the authenticity of messages. In the improbable event that all of the other parties conspire to defraud the monitor, the monitor will still know whether a message is authentic or not but will be unable to persuade impartial (and uninvolved) observers that he is telling the truth. In other words, the worst that could happen, from the

monitor's standpoint, for this authentication scheme is that he could, with low probability, find himself in the same situation that he was faced with with certainty in the system described earlier. It should be noted that the three participants need not use the same cryptoalgorithm, nor the same key sizes, etc. All that is required is that each provide a two-key authentication channel and share their decryption key with all other participants.

The resulting system provides authentication without secrecy with the capability to arbitrate host (transmitter) and monitor (receiver) disputes in an authentication scheme that is only computationally secure, i.e., the security is no better than the concealing cryptoalgorithm is secure. The essential dichotomy in authentication schemes illustrated by this application hinges on the question of whether the transmitter and receiver trust each other unconditionally or not. We choose to characterize this division by terminology that suggests the consequences of a failure of this trust.

with arbitration -- without arbitration

Simmons has recently constructed several classes of provably secure authentication codes that permit arbitration of transmitter/receiver disputes called $\mathbf{A}^2$ codes [19,20]. The relationship between these $\mathbf{A}^2$ codes and the computationally secure cryptographic based authentication with arbitration schemes just described is almost exactly the same as the relationship of the provable secure authentication codes, **A** codes, described earlier and the related computationally secure cryptographic based authentication schemes. For the sake of completeness, we exhibit a provably secure authentication with arbitration code for a one-bit source that is the extension of the Cartesian product construction of an authentication code described earlier. Clearly, in any authentication scheme there must be some uncertainty as to what messages will be accepted and/or certified by the arbiter for any participant whose cheating is to be prevented. In particular, for $\mathbf{A}^2$ codes this means that the opponent (outsider) must be uncertain as to which messages the receiver will accept (the opponent doesn't care whether the arbiter will later certify an accepted message as having come from the transmitter or not); however, the receiver (insider) must be uncertain as to which messages the arbiter will certify and the transmitter (insider) must also be uncertain as to which messages the receiver will accept. It turns out that there are infinite families of $\mathbf{A}^2$ codes in which all of these uncertainties to the various potential cheaters can be made to be the same and hence in which the resulting codes are perfect in the sense of channel usage described earlier. Our example is the smallest of these codes possible -- one bit of information is communicated in the message and one bit of uncertainty is presented to insiders and outsiders alike. Consider the Cartesian construction for authenticating rules:

$$A = \begin{pmatrix} H\ H\ -\ - \\ H\ -\ H\ - \\ -\ H\ -\ H \\ -\ -\ H\ H \end{pmatrix} \otimes \begin{pmatrix} T\ T\ -\ - \\ T\ -\ T\ - \\ -\ T\ -\ T \\ -\ -\ T\ T \end{pmatrix}$$

or

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

where the source is a toss of a fair coin, H or T. The authentication with arbitration protocol calls for the receiver to choose one of the authenticating rules, a_i, with a uniform probability distribution. For example, a_1, says that a head outcome to the transmitter's coin toss could be communicated by the transmitter using either message m_1 or m_2. Similarly, messages m_5 or m_6 would communicate source state "tails", while messages m_3, m_4, m_7 and m_8 would be rejected by the receiver under rule a_1 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 for each state of the source. The receiver communicates his choice of an authenticating rule to the arbiter in secret (from the transmitter and the opponent(s)). According to the protocol, the receiver has committed himself to accepting as authentic precisely those four messages corresponding to the source states in the authenticating rule he chose and to rejecting the remaining four as unauthentic. The arbiter randomly chooses one of the messages from the pair that would communicate "heads" and one from the pair that would communicate "tails" to form an encoding rule which he then forwards (in secret from the opponent and the receiver) to the transmitter. In this particular example for each choice of an authenticating rule there are four possible encoding rules, one of which is chosen by the arbiter with a uniform probability distribution. It is also the case that each possible encoding rule occurs in four different, but equally likely, authenticating rules so that the transmitter is uncertain as to the authenticating rule chosen by the receiver even though he knows two messages (one communicating source state "heads" and one communicating source state "tails") used in that rule. For example, assume that the receiver chose a_1, and that the arbiter constructed the encoding rule:

$$e = \begin{array}{c|cccccccc} & m_1 & m_2 & m_3 & m_4 & m_5 & m_6 & m_7 & m_8 \\ e & - & H & - & - & T & - & - & - \end{array} .$$

Source state H would be communicated under the established protocol by the transmitter sending message m_2 while T would be communicated by sending message m_5. According to the protocol, m_2 and m_5 will not only be accepted as "authentic" by the receiver, but will also 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 arbiter would not certify either of these messages as ones the transmitter could have sent under the established protocol.

Using this authentication scheme we now show that the immunity provided to each of the five types of cheating described earlier is to hold the cheater to a probability of 1/2, i.e., one bit of protection, as claimed irrespective of which type of cheating is considered. The easiest of the deceptions to analyze is the case of the outsider (opponent) who only knows the "system", i.e., he knows what the procedures are but does not know the receiver's or arbiter's choices. It should be clear that if he attempts to impersonate the transmitter and send a message when none has been sent, his probability of choosing one of the four (out of eight) messages that the receiver has agreed to accept (in his choice of an encoding rule) is 1/2 since in each case 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, say m_1, 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 these four leave him with four equally likely possibilities for the message that the transmitter is to use to communicate the other state of the source, and much more importantly, 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 substituting a message that the receiver will accept as communicating the other state of the source is still 1/2.

Consider next the case of the transmitter disavowing a message that he actually sent. In order to succeed, the transmitter must not only choose a message that the receiver will accept, but also one that is not used in the encoding rule forwarded by the arbiter. In other words he must choose a message that was used in the authenticating rule that the receiver chose, but not used in the encoding rule generated by the arbiter's choice. Continuing with the example used above, the transmitter knows from the encoding rule that was given to him by the arbiter that the receiver must have chosen one of the four authenticating rules:

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	-

Since messages m_3 and m_8 do not appear in any of these rules, the transmitter can be certain that either of these messages would be rejected by the receiver as unauthentic. Each of the remaining four messages, m_1, m_4, m_6, and m_7, appears in two of the equally likely choices of an authenticating rule, hence he cannot do better than choose one out of these four messages with equiprobability. 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 disavow having sent it, since he knows that the arbiter will not certify it as a message that would have been used under the established protocol.

Finally, we consider the two types of cheating available to the receiver. Of the four messages that he has agreed (with the arbiter) that he will accept as authentic, since they are used in his choice of an authenticating rule, two will be certified as being messages that could have been used under the established protocol and two will not be certified. The receiver will succeed in fraudulently attributing a message to the transmitter if he is able to choose one of the pair 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 selection procedure chooses among the acceptable (to the receiver) messages with a uniform probability distribution. If he waits until he receives a message from the transmitter communicating a source state, say m_2, indicating that the outcome of the coin flip was heads, he is still totally uncertain as to which of the messages that could be used to communicate the source state "tails" will be certified by the arbiter. The result however is that either message m_5 or m_6 is equally likely to be the one that will be certified, and his probability of successfully substituting a message conveying a different state of the source than was communicated in the message sent by the transmitter, i.e., of substituting one which will both communicate a different state of the source and will subsequently be certified by the arbiter as a message the transmitter could have sent under the established authentication protocol is 1/2.

This example illustrates all of the essential features of authentication codes that permit arbitration, A^2-codes. Three bits of information had to be communicated to specify one of eight equally likely messages. According to the protocol, this communication provides one bit of information (to the receiver) about the source state, one bit of protection (to the transmitter and receiver) against deception by outsiders and one bit of protection (to the transmitter or to the receiver) against cheating by insiders. Since the probability of success for the "cheater" in all cases is simply the probability that a randomly chosen message will be successful (in cheating) it seems reasonable to describe the code illustrated here as perfect. As has been pointed out in earlier papers on authentication codes without arbitration, these "perfect" codes are also perfect in the natural sense that all of the information transmitted is used either to communicate the state of the source or else to confound one of the cheating parties.

Conclusion

The purpose of the lengthy description of the various authentication problems and schemes given here was two-fold: first to persuade the reader that each of the three essential dichotomies is genuine and arises in real-world situations and secondly, by construction of solutions to show that these really are independent classifications of authentication schemes. Figure 1 summarizes the resulting classification and provides one or more examples of schemes illustrating each basic class. For the more recent contributions to authentication theory, the principal authors and/or collaborating coauthors are also indicated.

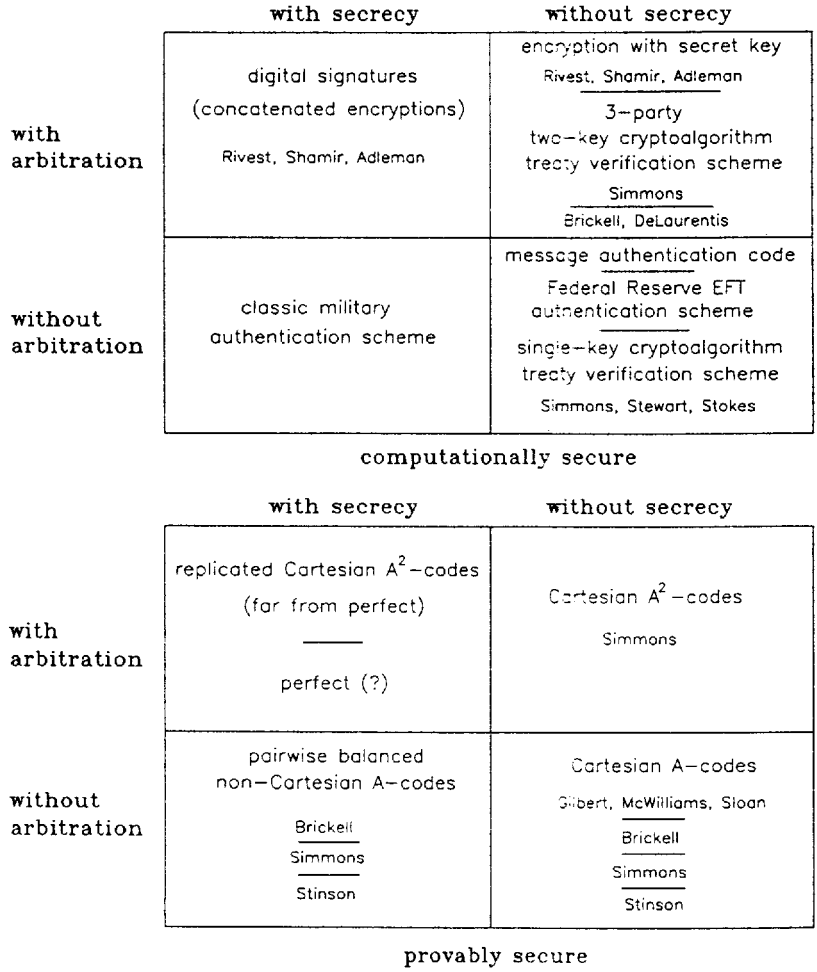

Figure 1. The Natural Taxonomy of Authentication Schemes

References

1. D. Chaum, "Security without identification: Transaction systems to make big brother obsolete," Communications of the ACM, Vol. 28, Oct. 1985, pp. 1030-1044.

2. A. Fiat and A. Shamir, "How to prove yourself: Practical solutions to identification and signature problems," Presented at Crypto'86, Santa Barbara, CA, Aug. 11-15, 1986, pp. 18-1 thru 18-7 of the Conference Abstracts and Papers.

3. O. Goldreich, S. Micali and A. Wigderson, "Proofs that yield nothing but their validity and a methodology of cryptographic protocol design," In The Computer Society of IEEE, 27th Annual Symp. on Foundations of Computer Science (FOCS), pp. 174-187, IEEE Computer Society Press (1986). Toronto, Ontario, Canada, Oct. 27-29, 1986.

4. P. D. Merillat, "Secure stand-alone positive personnel identity verification system (SSA-PPIV)," Sandia National Laboratories Tech. Rpt. SAND79-0070 (March).

5. G. J. Simmons, "A system for verifying user identity and authorization at the point-of-sale or access," Cryptologia, Vol. 8, No. 1, January 1984, pp. 1-21.

6. C. L. Henderson, A. M. Fine," Motion, intrusion and tamper detection for surveillance and containment," Sandia National Laboratories Tech. Rpt. SAND79-0792 (March 1980).

7. G. J. Simmons," A game theory model of digital message authentication," Congressus Numerantium 34 (1982), pp. 413-424.

8. G. J. Simmons, "Message authentication: A game on hypergraphs," Proceedings of the 15th Southeastern Conference on Combinatorics, Graph Theory and Computing, Baton Rouge, LA, March 5-8, 1984, pp. 161-192.

9. G. J. Simmons, "Authentication theory/coding theory," Proceedings of Crypto'84, Santa Barbara, CA, August 19-22, 1984, in Advances in Cryptology, Ed. by R. Blakley, Springer-Verlag, Berlin (1985), pp. 411-432.

10. G. J. Simmons, "The practice of authentication," Proceedings of Eurocrypt'85, Linz, Austria, April 9-11, 1985, in Advances in Cryptology, ed. by Franz Pichler, Springer-Verlag, Berlin (1986), pp. 261-272.

11. E. N. Gilbert, F. J. MacWilliams, N.J.A. Sloane, "Codes which Detect Deception," The Bell System Tech. Journal, Vol. 53, No. 3, March 1974, pp. 405-424.

12. E. F. Brickell, "A Few Results in Message Authentication," Proceedings of the 15th Southeastern Conference on Combinatorics, Graph Theory and Computing, Baton Rouge, LA, March 5-8, 1984, Congressus Numerantium, Vol. 43, Dec. 1984, pp. 141-154.

13. D. R. Stinson, "Some Constructions and Bounds for Authentication Codes," presented at Crypto'86, Santa Barbara, CA, Aug. 12-15, 1986, to appear in Journal of Cryptology, 1987.

14. D. R. Stinson, "A Construction for Authentication/secrecy Codes from Certain Combinatorial Designs," to appear in Journal of Cryptology.

15. Dept. of the Treasury Directive, "Electronic funds and securities transfer policy -- message authentication," Aug. 16, 1984, signed by Donald T. Regan, Secretary of the Treasury.

16. G. J. Simmons, "Verification of treaty compliance -- revisited," Proceedings of the IEEE Computer Society 1982 Symposium on Security and Privacy, Oakland, CA, Apr. 25-27 (1983), pp. 61-66.

17. G. J. Simmons, R. E. D. Stewart, P. A. Stokes, "Digital data authenticator," Patent Application SD2654, S42640 (June 30, 1972).

18. C. H. Meyer and S. M. Matyas, Cryptography: A New Dimension in Computer Data Security, John Wiley & Sons, New York (1982).

19. G. J. Simmons, "A Cartesian Product Construction for Authentication Codes that Permit Arbitration," to appear in J. of Cryptology.

20. G. J. Simmons, "Authentication Codes that Permit Arbitration," Proc. of the 18th Southeastern Conference on Combinatorics, Graph Theory and Computing, Boca Raton, FL, Feb. 23-27, 1987.