THE PRISONERS' PROBLEM AND THE SUBLIMINAL CHANNEL[†]

Gustavus J. Simmons

Sandia National Laboratories
Albuquerque, NM 87185

INTRODUCTION

Two accomplices in a crime have been arrested and are about to be locked in widely separated cells. Their only means of communication after they are locked up will be by way of messages conveyed for them by trustees -- who are known to be agents of the warden. The warden is willing to allow the prisoners to exchange messages in the hope that he can deceive at least one of them into accepting as a genuine communication from the other either a fraudulent message created by the warden himself or else a modification by him of a genuine message. However, since he has every reason to suspect that the prisoners want to coordinate an escape plan, the warden will only permit the exchanges to occur if the information contained in the messages is completely open to him -- and presumably innocuous. The prisoners, on the other hand, are willing to accept these conditions, i.e., to accept some risk of deception in order to be able to communicate at all, since they need to coordinate their plans. To do this they will have to deceive the warden by finding a way of communicating secretly in the exchanges, i.e., of establishing a "subliminal channel" between them in full view of the warden, even though the messages themselves contain no secret (to the warden) information[‡]. Since they anticipate that the warden will try to deceive them by introducing fraudulent messages,

[†] This work performed at Sandia National Laboratories supported by the U. S. Department of Energy under contract number DE-AC04-76DP00789.

[‡] This description is strictly true, i.e., the secret communication is not by way of a coded selection of messages or of prearranged code words appearing in a message, etc.

they will only exchange messages if they are permitted to authenticate them.

Even though the problem description appears paradoxical, it has a surprisingly simple solution. In this paper we describe one way to realize such a subliminal channel, i.e., a solution to the prisoners' problem.

AUTHENTICATION WITHOUT SECRECY

Superficially, the problem resembles the "authentication without secrecy" problem discussed by the author in a series of earlier papers [1,2,3]. While it is true that the prisoners (transmitter/receiver) insist on authentication and the warden (opponent) demands access to the message content so that communication takes place over an authentication without secrecy channel, the subliminal channel is actually a subversion (by the transmitter/receiver), of the authentication channel. In order to appreciate how the subversion is accomplished, we must briefly describe the authentication without secrecy channel. The chapter entitled Message Authentication Without Secrecy [1] in Secure Communications and Asymmetric Cryptosystems is recommended for a more complete discussion.

Authentication depends on the transmitter introducing prearranged redundant information into the messages, the presence of which in a decrypted cipher will be interpreted by the receiver as indicating that the communication is genuine. For example, authentic messages may be required to end in a specified (minimum) number of zeroes [4] or in a particular suffix as is the common practice in military authentication systems. Conversely, the absence of the prearranged redundant information is interpreted to mean that the communication is not genuine. Since the opponent must be prevented from simply "stripping off" the authenticating information from one message and appending it to another, the authentication is generally secured by encryption. In many instances the message along with the authenticating information is block encrypted into a cipher using either a single key or a two key cryptoalgorithm. In either case, if the cryptoalgorithm is adequately secure, the probability, P_A, of an opponent successfully choosing a cipher that will be accepted by the receiver as a genuine message is simply related to the information content, H_r, of the redundant authenticating information: optimally $P_A = 2^{-H_r}$.

The host to the communication channel (the warden) satisfies himself that nothing has been concealed in the message by decrypting the cipher -- either with an encryption/decryption key given to him after the exchange has taken place if a single key cryptoalgorithm was used or with a decryption key given to him in advance of the exchange in a two key system. For single key cryptographic systems, the host must "trust" the transmitter/receiver until he receives the decryption key corresponding to the last cipher exchange -- which if the message is very long may involve an unacceptable level of risk (to him) of covert communication. There is no way of avoiding this problem for single key systems however, since if the host has the key in advance so that he can decrypt the cipher, he could also encrypt and hence create an undetectable forgery. The essential -- and vital -- difference for two key cryptosystems is the absence of this need for even a temporary "trust" by either party or the other since the host can have the decryption key in his possession prior to any exchange of messages, and hence have the ability to verify the message content prior to forwarding the cipher. On the other hand, since the host cannot infer the unknown encryption key, the transmitter/receiver are confident that he cannot better his guessing odds of choosing an acceptable cipher. Actual authentication without secrecy channels are frequently much more complex than this simplified description suggests. For example, in an early single key version of a system to authenticate data from unmanned seismic stations designed to monitor Russian compliance with a commprehensive test ban treaty, the messages, which were extremely long data streams, were to be transmitted in the clear along with an appended -- much shorter -- function of the entire message.

The essential points to an authentication without secrecy channel are that;

a) the receiver authenticates a message through the presence of H_r bits of redundant -- expected -- information in the decrypted cipher,

b) the host to the communication channel verifies that nothing has been concealed by decrypting the ciphers and verifying that the resulting message is precisely what he expected based on a foreknowledge of the message.

As mentioned before, the system is operationally different for the host depending on whether the cryptoalgorithm is single or two key, since this determines whether he can check for concealed information before or after the exchange occurs. However, this does not alter the way in which he satisfies himself that nothing is concealed -- namely, that the cipher decrypts to the expected message.

THE SUBLIMINAL CHANNEL

In order to communicate H_m bits of information with H_r bits of authentication, $H_m + H_r$ bits in total must be exchanged. The notion of a subliminal channel in such a situation is extremely simple.

Conceivably the transmitter/receiver could give up some of their ability to authenticate -- without the host being aware of this -- and use the resulting information capacity, say H_s bits, to communicate secretly. Admittedly, in such a case, if the host tried to deceive the receiver sufficiently many times, and if he was told in each case whether he succeeded or not, he would eventually find that his probability of success was $2^{-(H_r-H_s)}$ rather than 2^{-H_r} as would be the case for the authentication channel agreed to in advance by the host and the transmitter/receiver. We first illustrate these notions with the smallest simplified single key example possible. $H_m = H_s = H_r - H_s = 1$, in other words, the messages consist of a single bit of information, the outcome of a fair-coin toss for example, the subliminal channel will have a capacity of one bit subverted from a two bit -- one chance in four of deception -- authentication channel to leave a single bit authentication channel.

We first describe an example of an authentication without secrecy channel that allows a one bit message to be authenticated with two bits, i.e., the opponent's probability of deceiving the receiver is $2^{-H_r} = 1/4$. Our encoding rule relating the outcome of a fair-coin toss to three bit messages is simple; Heads = even parity, Tails = odd parity, Figure 1. In other words, if the outcome of the coin toss is Heads, the warden would only allow an even parity exchange to occur. Actually, what the warden and the prisoners would have agreed to in advance would be a key list or in actual practice an equivalent functional description of a partitioning of the message space by keys such as is tabulated in Figure 2.

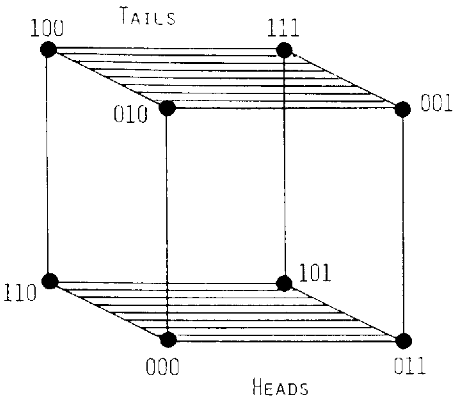


Figure 1

	Heads	Tails
1.	000	100
2.	000	010
3.	000	001
4.	000	111
5.	011	100
6.	011	010
7.	011	001
8.	011	111
9.	101	100
10.	101	010
11.	101	001
12.	101	111
13.	110	100
14.	110	010
15.	110	001
16.	110	111

Figure 2. Keys for a 1-bit channel and a
2-bit authentication without secrecy

After the prisoners have been locked up, having selected in secret from the warden one of the sixteen possible keys, there are two ways the warden might attempt to deceive one of them.

a) before the critical coin toss occurs, he could deliver a message of his own choice,

b) after the coin toss has occurred and the prisoner who is acting as the transmitter gives to the warden a message to communicate its outcome, the warden could substitute another message.

Even in the most general scheme, the opponent has only these two types of deceit available as his options; (a) impersonation or (b) substitution.

Clearly, if the sixteen keys are equiprobable, the opponent's probability of successful impersonation will simply be the probability that he "guesses" a message that is in the secret key, i.e., 4/16 = 1/4 in this example. On the other hand, if he waits until after the coin toss has occurred, whichever message he is given lies in exactly four keys, corresponding to each of the messages of opposite parity. In this case, his probability of successful substitution is simply the probability that he "guesses" the unique message of opposite parity in the secret key, i.e., 1/4 again. In other words, in this example the opponent's chances of success are 1/4 irrespective of whether he tries impersonation or substitution. Figure 3 shows the key space as a superposition of the message encoding rule and the authentication keys in a way that suggests how the problem generalizes to the more general case.

In order to set up a one bit subliminal channel, the prisoners select a secret key, not from the list of sixteen that they ostensibly choose from, but rather from among the eight keys shown in Figure 4.

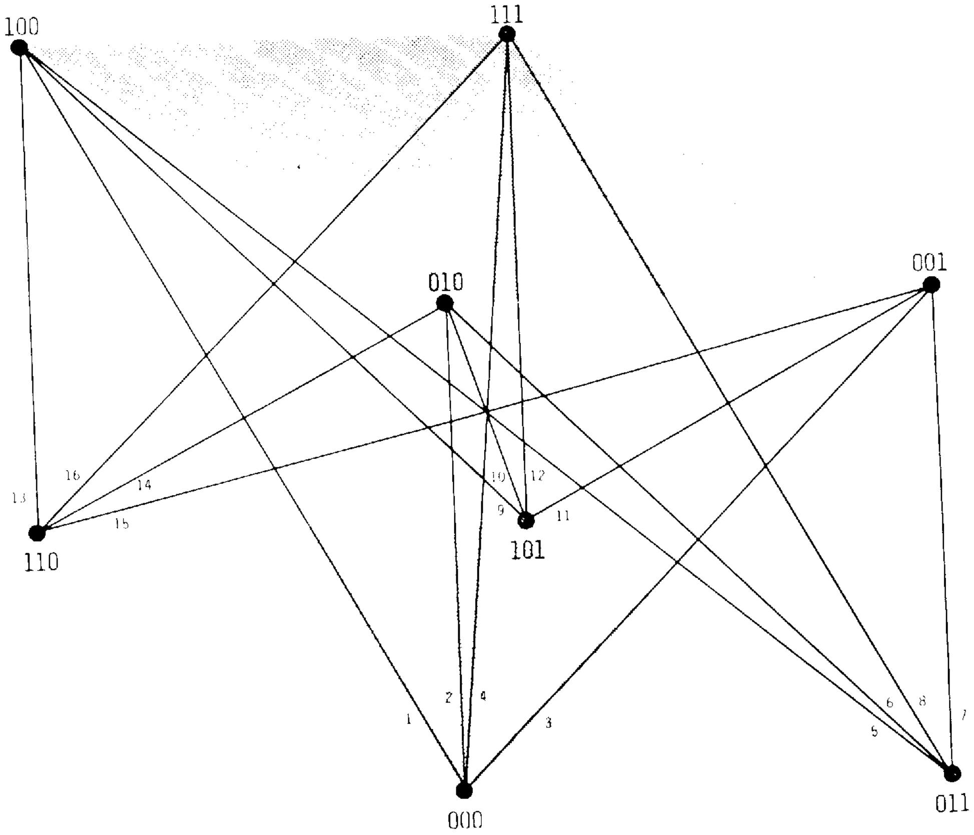


Figure 3. Superimposed encoding rules (shaded regions) and authentication keys (edges).

1.	000	011	111	100
2.	000	011	010	001
3.	000	110	010	100
4.	000	110	111	001
5.	101	110	111	100
6.	101	110	010	001
7.	101	011	010	100
8.	101	011	111	001

Figure 4. Keys for a 1-bit overt channel, a 1-bit subliminal channel and a 1-bit authentication without secrecy channel.

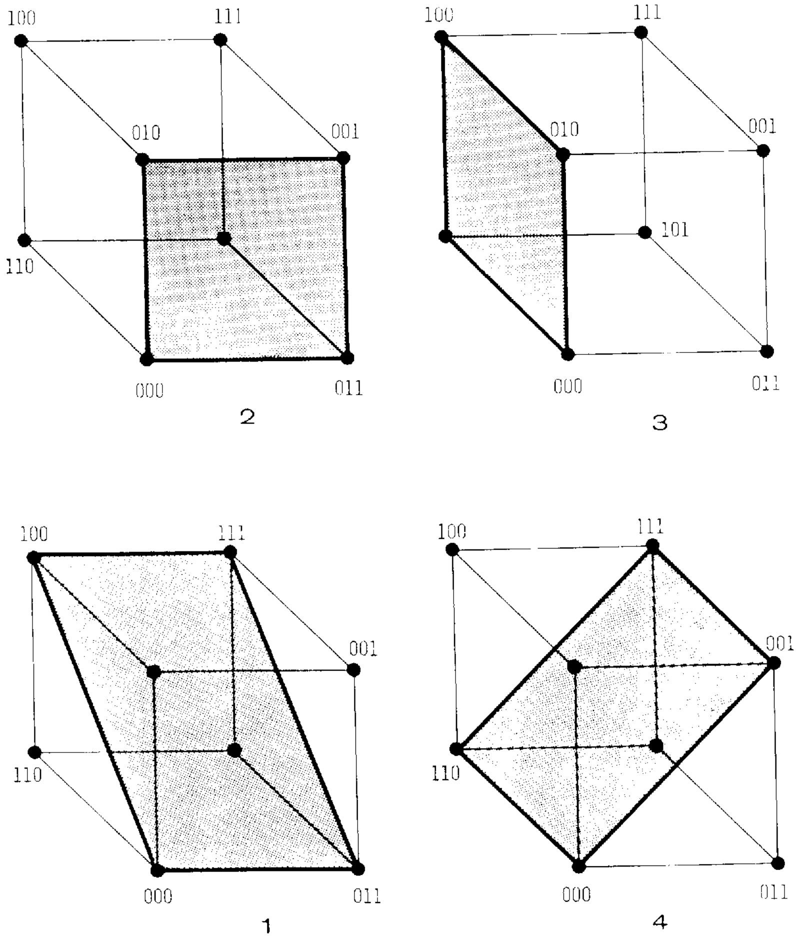


Figure 5

Each of the keys in the list of Figure 4 includes four from the list in Figure 2, but what is more important each message is in precisely four of the keys in Figure 4. For example, the message 000 is in keys 1, 2, 3 and 4 as shown in Figure 5. Figure 6 shows the eight authentication/subliminal channel keys in a schematic way similar to that shown in Figure 3 for the eight authentication keys.

Since the receiver will now accept (unbeknown to the opponent) any of four different messages, two of odd parity and two of even parity, the opponent's probability of successful impersonation is 2/4 = 1/2. Similarly, his probability of successful substitution is 2/4 = 1/2. As mentioned earlier, if he could make sufficiently many attempts at deception with this scheme, with feedback on each attempt as to whether he had succeeded or not, he could quickly infer that something was amiss when the estimate of his probability of success converged to 1/2 rather than to 1/4. This is not a problem in actual applications through since the probability of success is chosen to be very very small so that it is infeasible to estimate it by testing of the channel. Furthermore the simplified model used here doesn't provide the essential capability of key changes with successive messages, an essential property if the opponent is to be denied the option of simply substituting a stale message from some previous exchange.

Figure 7, shows two planes θ and φ(subsets of messages); θ on the messages 000, 010, 111, 101 and φ on 100, 110, 011, 001. Each key in Figure 4 intersects each of these planes, θ or φ, in exactly two messages -- one of odd parity and one of even. Hence if the transmitter receiver have agreed that a message in θ say communicates a subliminal 1 while a message in φ communicates a 0, then no matter which key from Figure 4 they chose, and irrespective of whether the outcome of the coin toss in Heads or Tails, they can send a subliminal bit. For example, say they choose key 6, then messages 110 and 001 are in φ and 010 and 101 are in θ. If Heads is to be transmitted, either message 110 or 101 will be correctly interpreted by the receiver (and the host) to mean that Heads was the outcome of the coin toss. In addition, the receiver will interpret 110, which is in φ, to be a subliminal 0 or 101, which is in θ, to be a subliminal 1. It should be an easy matter for the reader to convince himself, using Figures 6 and 7, that there is nothing special about this particular choice of key and message bit, and that the same is true for all possible choices of keys, encoding rules and subliminal bits. In fact, this small -- eight message -- example was constructed from the (8,4,2,3) orthogonal array: others of arbitrary size -- numbers of messages and keys -- can be constructed similarly.

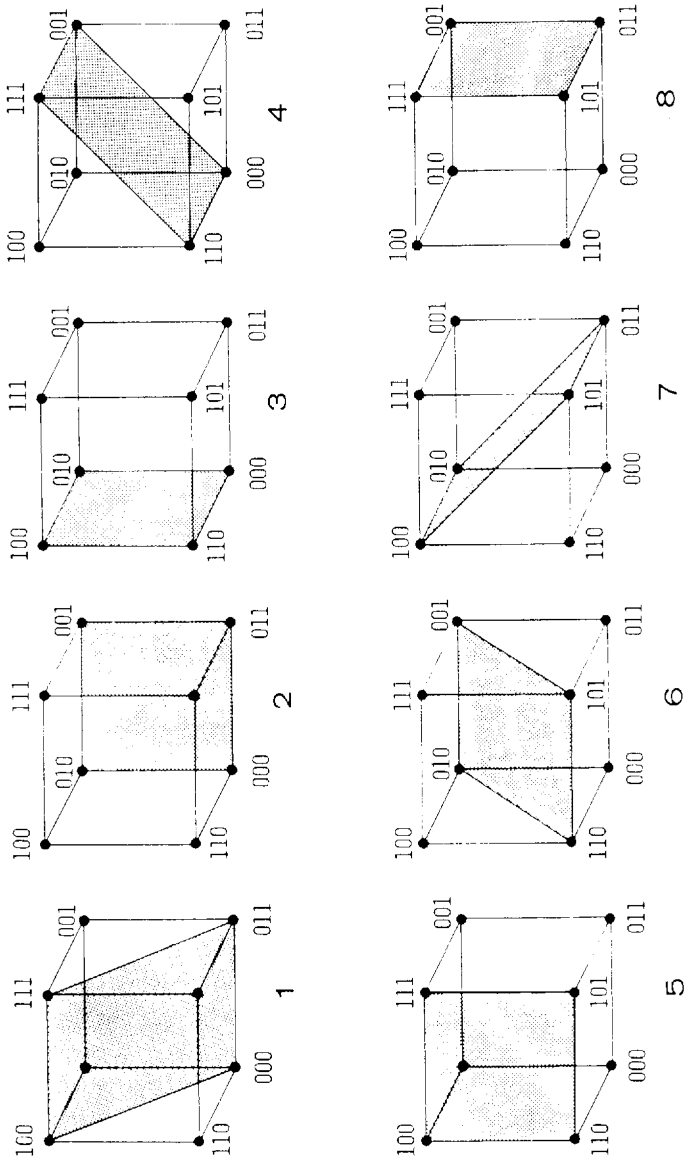


Figure 6

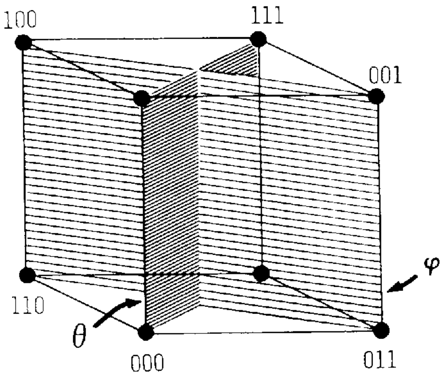


Figure 7

In this section we describe a single bit subliminal channel that is as cryptosecure as a product of three large primes is difficult to factor, and whose existence is as hard to detect as the associated factoring problem. In order to explain the subliminal channel, we must first collect a few elementary number theory results.

Let $n = pqr$ be the product of three primes p, q and r where $p \equiv q \equiv 3 \pmod 4$, $r \equiv 5 \pmod 8$ and n is computationally infeasible to factor; i.e., the same conditions are observed on the choices of p, q and r as would be necessary for the choices for "good" p and q in the RSA cryptosystem [5], however, as will be apparent in the subsequent discussion we are only using the factoring problem as a means of constructing a one-way function -- not as a basis for a cryptoalgorithm as in the RSA cryptoscheme. Any quadratic residue x^2 in the ring, R_n, of residues mod n, where $(x^2,n) = 1$, has eight square roots of the form:

$$x = (\pm\alpha)qr + (\pm\beta)pr + (\pm\gamma)pq \qquad (1)$$

where $(\pm\alpha)$ denotes $0 \leqslant \alpha < p$ and $p-\alpha$, etc.

$$x^2 \equiv \alpha^2q^2r^2 + \beta^2p^2r^2 + \gamma^2p^2q^2 \pmod n \quad . \qquad (2)$$

As is well known, if one can extract square roots in R_n, one can also, with probability that goes to one like $\frac{1}{2}-2^{-k}$ for k attempts, factor n. In other words, given only a square, x^2, in R_n and n, it will with probability one be computationally infeasible to find any square root of x^2. On the other hand, as we show below, if one knows p, q and r it is only of order log n computational difficulty to compute the eight square roots of x^2.

(2) can be replaced by three simple quadratic congruences in terms of p, q and r:

$$\left.\begin{aligned} \alpha^2(q^2r^2) &\equiv x^2 \quad (\text{mod } p) \\ \beta^2(p^2r^2) &\equiv x^2 \quad (\text{mod } q) \\ \gamma^2(p^2q^2) &\equiv x^2 \quad (\text{mod } r) \end{aligned}\right\} . \tag{3}$$

If one knows p, q and r, i.e., the factorization of n, the residue of the parenthetic squares of the products of pairs of prime factors can be calculated with respect to the modulus in advance and by one application of the Euclidean algorithm the multiplicative inverse found. Let $(x)_p$ denote the least positive residue of x (mod p) then (3) can be rewritten in the form;

$$\left.\begin{aligned} \alpha^2 &\equiv \left((q^2r^2)_p^{-1}x^2\right)_p \equiv a(x) \quad (\text{mod } p) \\ \beta^2 &\equiv \left((p^2r^2)_q^{-1}x^2\right)_q \equiv b(x) \quad (\text{mod } q) \\ \gamma^2 &\equiv \left((p^2q^2)_r^{-1}x^2\right)_r \equiv c(x) \quad (\text{mod } r) \end{aligned}\right\} . \tag{3'}$$

Consequently finding the eight square roots of x^2, given the factorization of n is only computationally as difficult as solving the three congruences in (3'). As is well known [6], the quadratic congruence

$$x^2 \equiv a \quad (\text{mod } p) \tag{4}$$

has the simple solution

$$x \equiv \pm\, a^{k+1} \quad (\text{mod } p) \tag{5}$$

if p = 4k+3. The computational difficulty is therefore proportional to log p in this case. Less well known, is the fact that if the prime is of the form p = 8k+5 congruence (4) can also be solved by exponentiation at a cost proportional to log p by (5) if

$$a^{2k+1} \equiv 1 \quad (\text{mod } p) \tag{6}$$

or else by

$$x \equiv \pm\left(\frac{p+1}{2}\right)(4a)^{k+1} \quad (\text{mod } p) \tag{7}$$

if

$$a^{2k+1} \equiv -1 \qquad (\mathrm{mod}\ p) \quad . \qquad (8)$$

Combining the results of (1), (3'), (5) and (7) we see that the comp tational difficulty in computing the eight square roots of x^2 in R_n given the factorization of n is only proportional to log p; i.e., a computationally easy task for values of n sufficiently large that the computation of the square roots would be completely infeasible without the factorization.

We next explain the conditions imposed earlier on the primes p, q and r; namely that $p \equiv q \equiv 3 \pmod 4$ and $r \equiv 5 \pmod 8$. The eight square roots of x^2 in R_n, where n = pqr in this case, are grouped so that precisely two satisfy each of the four possible pairs of conditions $x < n/2$ or $x > n/2$ and $\left(\frac{x}{n}\right)$† = +1 or $\left(\frac{x}{n}\right) = -1$. To show this we first note that for a p a prime of form p = 4k+3 x and its complement p-x have different Jacobi symbols:

$$\left(\frac{x}{p}\right) = \left(\frac{x-p}{p}\right) = \left(\frac{-1}{p}\right)\left(\frac{p-x}{p}\right) = -\left(\frac{p-x}{p}\right)$$

On the other hand, for a prime p of the form p = 4k+1, x and its complement p-x have the same Jacobi symbol:

$$\left(\frac{x}{p}\right) = \left(\frac{x-p}{p}\right) = \left(\frac{-1}{p}\right)\left(\frac{p-x}{p}\right) = \left(\frac{p-x}{p}\right) \ .$$

Therefore the Jacobi symbols for a square root x of x^2 in R_n and its complement n-x are the same:

$$\left(\frac{x}{pqr}\right) = \left(\frac{x-pqr}{pqr}\right) = \left(\frac{-1}{pqr}\right)\left(\frac{pqr-x}{pqr}\right) = \left(\frac{-1}{p}\right)\left(\frac{-1}{q}\right)\left(\frac{-1}{r}\right)\left(\frac{pqr-x}{pqr}\right)$$

$$= (-1)^2(+1)\left(\frac{pqr-x}{pqr}\right) = \left(\frac{pqr-x}{pqr}\right) \quad .$$

Since for each square root x, both x and n-x are square roots of x^2 in R_n, the square roots partition into four < n/2 and four > n/2.

For any odd modulus n for which there exists at least one prime p for which‡

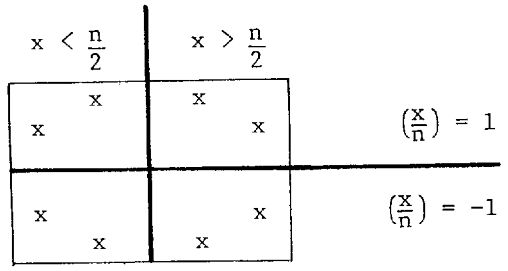
$$p^{2k+1} \| n$$

there are $\varphi(n)/2$ reduced residues x for which $\left(\frac{x}{n}\right) = +1$ and $\varphi(n)/2$ for which $\left(\frac{x}{n}\right) = -1$. This result combined with that of the preceding

† (x/n) is the Jacobi symbol.

‡ $p^{2k+1} \| n$ means that p^{2k+1} divides n but p^{2k+2} does not.

paragraph implies that the eight square roots of x^2 group into pairs as claimed: figuratively



$x < \frac{n}{2}$
$x > \frac{n}{2}$
x
x
x
x
$\left(\frac{x}{n}\right) = 1$
x
x
x
x
$\left(\frac{x}{n}\right) = -1$

Table 1 shows this splitting for the forty-eight square roots in R_{105} for example.

Table 1.

	square roots			
	$x < \frac{n}{2}$		$x > \frac{n}{2}$	
squares	$\left(\frac{x}{n}\right) = 0$	$\left(\frac{x}{n}\right) = 1$	$\left(\frac{x}{n}\right) = 0$	$\left(\frac{x}{n}\right) = 1$
1	29 34	1 41	71 76	64 104
4	37 47	2 23	58 68	82 103
16	11 31	4 46	74 94	59 101
46	19 44	16 26	61 86	79 89
64	22 43	8 13	62 83	92 97
79	17 38	32 52	67 88	53 73

Using the properties of quadratic congruences in R_{pqr} just established, it is possible to define a subliminal channel -- whose very existence is computationally infeasible for an opponent to establish. Let **X** be the set of quadratic residues in R_n. The authentication without secrecy channel that we first define has **X** as the set of acceptable (authentic) messages and the square roots of the $x^2 \varepsilon \mathbf{X}$ as the ciphers to be exchanged. At the time that the channel is agreed to by the host and the transmitter/receiver, the transmitter/receiver give to the host the modulus n and declare in advance the Jacobi symbol and magnitude bound they have agreed to accept as the authentic, square root. They do not tell the host that the modulus is the product of three primes -- a fact that is as hard for him to verify as it would be for him to factor n. The host can of course at a cost proportional to log n, verify that n is composite. He therefore knows that for every

$x^2 \in \mathbf{X}$ there is a square root satisfying the preannounced criteria for authentication. However it is completely infeasible for him to establish that there is more than one such square root -- irrespective of whether he suspects this to be the case or not. The subliminal channel requires that both the transmitter and the receiver do O(log n) computation. Given the message $x^2 \in \mathbf{X}$, the transmitter calculates the eight square roots of x^2, selects the pair that satisfy the authentication criteria and then sends the smaller of these roots to communicate a subliminal 0 or the larger to communicate a 1. The host, knowing x^2 and having been given x by the transmitter squares the cipher (x) to verify that it corresponds to the already known message and also computes $\left(\frac{x}{n}\right)$ to verify that the cipher has the correct Jacobi symbol and magnitude. As far as he can determine this is a legitimate authentication without secrecy channel. The receiver, however, knowing p, q and r upon receiving x^2 solves for the eight square roots of x^2, etc., just as the transmitter did and decodes the subliminal bit by checking whether he received the larger or smaller "authentic" square root of x^2.

Example:

$$p = 103 \equiv q = 107 \equiv 3 \bmod 4$$

$$r = 101 \equiv \bmod 8$$

$$n = 103 \times 107 \times 101 = 1{,}113{,}121 \quad .$$

Let the message be $x^2 = 384{,}512$ so that (3') becomes

$$\left.\begin{aligned} \alpha^2 &\equiv 66(x^2)_p \equiv 34 \quad (\bmod 103) \\ \beta^2 &\equiv 47(x^2)_q \equiv 85 \quad (\bmod 107) \\ \gamma^2 &\equiv 47(x^2)_r \equiv 33 \quad (\bmod 101) \end{aligned}\right\} \quad . \qquad (3')$$

Using (5), we solve for α and β.

$$\alpha \equiv \pm(34)^{26} \equiv 72 \text{ or } 31 \quad (\bmod 103)$$

$$\beta \equiv \pm(85)^{27} \equiv 37 \text{ or } 70 \quad (\bmod 107)$$

Since

$$33^{25} \equiv 100 \quad (\bmod 101)$$

we must use (7) to solve for γ

$$\gamma \equiv \pm 51(21)^{13} \equiv 72 \text{ or } 29 \quad (\bmod 101) \quad .$$

Substituting these values for α, β and γ into (1) we obtain the square roots of 384,512

$x < \frac{n}{2}$	$x > \frac{n}{2}$	
369,503	843,406	$(\frac{x}{n}) = +1$
269,715	743,618	
400,319	1,039,537	$(\frac{x}{n}) = -1$
73,584	712,802	

Thus if the authentication criteria were $x < n/2$ and $(\frac{x}{n}) = -1$ and the message was 384,512, the cipher 400,319 would communicate a subliminal 1 while 73,584 would communicate a 0.

There is a serious problem with the subliminal channel just described. Although the warden can't detect that the prisoners are exchanging a secret bit in each message, he can with probability as close to 1 as he wishes deceive one of them into accepting a forgery as an authentic message. To do this he uses forward search [7] in the following way:

a) He selects a random $x \varepsilon R_n$, $x < n/2$ or $x > n/2$ depending on the authentication criteria announced by the prisoners.

b) If $(x,n) \neq 1$, he has found a factor of n and has broken the underlying cryptosystem. The probability of this happening is $\approx 1/p + 1/q + 1/r$.

c) If $(x,n) = 1$, he calculates $(\frac{x}{n})$.

d) If $(\frac{x}{n})$ satisfies the authentication criteria -- which will be the case with probability 1/2 -- he calculates x^2 and gives it to the receiver (prisoner) who will accept it as an authentic message. The associated subliminal bit will randomly be 1 or 0 with probability 1/2.

To stop the forward search attack by the host, it is necessary to restrict the set of acceptable messages so that the opponents' chances of a random "hit" are sufficiently low to be an acceptable risk for the transmitter/receiver. For example, if the primes are all $\approx$ m bit numbers (binary representation) so that $\lceil \log_2 n \rceil = 3m$, and the number of quadratic residues is $\varphi(n) = (p-1)(q-1)(v-1)$, the number of acceptable square roots will be $\varphi(n)/4 \approx 3m-2$. The quadratic residues that are in a specified residue class with respect to

a prime s, with say m bits in its binary representation, is one su
subclass. In this case the warden's probability of a hit will be
$2^{-(m+1)}$ so that it is computationally infeasible for him to search
out an acceptable x^2 by testing random x's, on the other hand, the
number of possible messages is $\approx 2^{2(m-1)}$, i.e., the information co
tent per message can be as much as 2m-2 bits. This is not necessa
ily a good choice for an acceptable message subset, but illustrate
the essential point about how forward search can be protected agai
by the transmitter/receiver.

CONCLUSION

Since the discussion of the two examples of subliminal channels has been lengthy and unavoidably detailed, we conclude with a succinct statement of the principal on which subliminal channels a
based. In an authentication without secrecy channel wherein the host satisfies himself that nothing is hidden in the exchange by decrypting the cipher to recover the expected message and the receiver authenticates messages by verifying that m bits of preagreed upon redundant information are present, the transmitter/receiver can exchange -- unbeknown to the host -- part or all of this ability to authenticate for a corresponding bit-for-bit capability of subliminal communication. A simplified statement of this conclusion is that it is feasible to design cryptoalgorithms in which several ciphers decrypt to the same message, and hence in which it is possible to communicate some side information, over and above that in the message itself, by way of the identity of the particular cipher used. In addition, in the second example, it was shown that such a subliminal channel can be made just as difficult to detect as the underlying cryptoalgorithm is difficult to break.

REFERENCES

1. G. J. Simmons, "Message Authentication Without Secrecy," in Secure Communications and Asymmetric Cryptosystems, (ed. by G. J. Simmons) AAAS Selected Symposia Series, Westview Press, Boulder, CO (1982), pp. 105-139.

2. G. J. Simmons, "Verification of Treaty Compliance -- Revisited, Proceedings of the 1983 IEEE Symposium on Security and Privacy, Oakland, CA (April 25-27, 1983), to appear.

3. G. J. Simmons, Symmetric and Asymmetric Encryption," Computing Surveys, Vol. 11, No. 4 (Dec. 1979), pp. 305-330.

4. H. Ong and C. P. Schnorr, "Signatures through Approximate Representations by Quadratic Forms," *Proceedings of the IEEE Workshop on Communications Security (CRYPTO'83)*, University of California, Santa Barbara, CA (August 26-30, 1983), to appear.

5. R. Rivest, A. Shamir and L. Adleman, "A Method for Obtaining Digital Signatures and Public-Key Cryptosystems," *Communications of the ACM 21*, 2 (February 1978), 120-126.

6. D. H. Lehmer, "Computer Technology Applied to the Theory of Numbers in *Studies in Number Theory* edited by W. J. LeVeque, M.A.A. Studies in Mathematics, Vol. 6, Prentice Hall (1969), 132-133.

7. G. J. Simmons and D. H. Holdridge, "Forward Search as a Cryptanalytic Tool," *Proceedings of the 1982 Symposium on Security and Privacy*, Oakland, CA (April 26-28, 1982), 117-128.

SESSION II
MODES OF OPERATION