The History of Subliminal Channels

Gustavus J. Simmons

Abstract—**In 1978 the United States was considering adopting a national security protocol designed to enable the U.S.S.R. to verify how many Minuteman missiles the United States had emplaced in a field of 1000 silos without revealing which silos actually contained missiles. For this protocol to have been acceptable to the U.S.S.R., the messages would have had to be digitally signed with signatures which the U.S.S.R. could verify were authentic, but which the United States could not forge. Subliminal channels were the discovery that these digital signatures could host undetectable covert channels. In general, any time redundant information is introduced into a communication to provide an overt function such as digital signatures, error detection and/or correction, authentication, etc. it may be possible to subvert the purported function to create a covert (subliminal) communications channel. This paper recounts the development of subliminal channels from their origins when only a couple of bits could be communicated covertly to today when potentially a couple of hundred bits can be concealed in signatures generated using the most popular digital signature schemes.**

Index Terms—**Communication system security, cryptography, data security, nuclear weapons, protocols, steganography, subliminal channels, treaty verification.**

WHAT I would like to do this afternoon, probably for the last time for me, is tell you in more detail than I have done on previous occasions how subliminal channels came to be discovered.

In the Carter administration (we're going back 20 years to 1976–80), the President had two major defense initiatives that he was determined to push through during his presidency. One of these was the ratification of the SALT 2 treaty, which depended critically on what was then a radical notion: That the United States and the Soviet Union would cooperate with each other to the extent that each party would make it possible for the other by national means (that's a euphemism for satellites) to verify the number of strategic (intercontinental) missiles that the other had in place. The primary object of the treaty was to limit the number of strategic missiles that each side could legitimately field. In order for the treaty to be acceptable, though, there had to be some means for each party to verify that the other was complying with its terms. So on the one hand this treaty depended on each party cooperating with the other to make that possible, but on the other it had to be assumed that either party would cheat if they could do so without risk of detection.

The other initiative of the Carter administration, you may recall, was what in retrospect seems a rather silly thing, but at the time seemed serious a scheme for making the land-based Minuteman missile system survivable against a first strike by intercontinental ballistic missiles. This was 1976—coincidentally, the time that public key cryptography came on the scene—when the accuracy of delivery of intercontinental missiles had improved to the point where it was no longer possible to make a missile silo survive a targeted nuclear warhead just by hardening it more. The accuracy of missile delivery systems had gotten to the point where they could with high probability destroy the missiles in individual silos, and MIRV (multiple warheads with terminal guidance) had made it feasible to target individual silos. As a matter of fact, it was a popular saying in the defense community at the time that "This is the last generation of land missiles," and subsequent developments have proven this statement to have been approximately true.

The scheme the Carter administration was pushing was often referred to in the popular press as a "missile shell game." In a Minuteman field they were going to prepare 1000 silos and have 100 missiles that would shuttle about amongst these emplacements. There were to be "transportainers"—great trucks—that would continuously and randomly move around visiting all of the silos in a field. The transportainers would back up to a silo, go through the motions of loading or unloading a missile, and then trundle at five miles an hour, like the shuttle transport, to another silo and repeat the procedure. It was even envisioned that they would take on a dummy load—perhaps of water in tanks—so that from the exterior it would be impossible to tell whether the load was dummy or real. The idea was that since even from close range it would be impossible to tell whether a missile was being put in or taken out of the silo, after a period of time any knowledge an enemy might have had at the beginning as to which silos had missiles in them would have been dissipated and their certainty about which silos were occupied and which were empty would have vanished. An opponent (the Soviet Union presumably) could only guess at whether a particular silo was occupied or not. Consequently, all 1000 silos would have had to be targeted in order to be confident of destroying all 100 of the Minutemen. Since this dilution of the effectiveness of a first strike wasn't considered to be cost effective, it was thought that this would ensure the survival of an adequate force for a retaliation.

So here we have these two competing and apparently mutually exclusive requirements. On the one hand the SALT 2 treaty required the United States to provide a way for the Soviet Union to verify how many silos were occupied. On the

This paper is a transcript of an invited lecture and is reprinted from G. J. Simmons, "The History of Subliminal Channels," in *Proc. 1st Int. Workshop on Information Hiding*, Isaac Newton Inst., Univ. of Cambridge, Cambridge, U.K., May 30–June 1, 1996, pp. 237–256.

The author, retired, was with the Sandia National Laboratories, Albuquerque, NM 87106 USA. He is now at P.O. Box 365, Sandia Park, NM 87047 USA.

Publisher Item Identifier S 0733-8716(98)01111-1.

other hand, there was this elaborate scheme to conceal which ones were occupied. Incidentally, this might have been done using statistical techniques as was later negotiated to verify the MRBM (medium range ballistic missile) treaty. The Russians could have been allowed to say "We want to have a look in those 20 silos" and then estimate on the basis of how many out of the 20 silos they had chosen had missiles in them the expected number of occupied silos in the field.

But the level of suspicion in those days was such that this wasn't acceptable. In order for the treaties to go forward, there had to be a deterministic scheme whereby the Soviet Union could exercise their right to challenge and see how many missiles were in the field. If this was done by opening the lids of the silos so they could see from their satellites which ones were occupied, all the uncertainty that had been generated over a long period of time by shuffling the missiles around amongst the silos would be lost. The result would have been that the United States would be back at ground zero, and it would be a long time before there had been sufficient potential moves by the transportainers that the Russians would be sufficiently uncertain again as to which silos were occupied.

This is the setting of the Carter administration's problem. They had a dilemma of the first order. On the one hand they needed to be able to compellingly respond to the Soviet Union's challenge as to how many Minuteman silos were occupied, but the survival of the force depended on not revealing whether any particular silo was occupied. The Department of Defense put up for bid to the defense contractor community in the United States a request for a solution to this; i.e., to devise a way you could compellingly convince the Russians of how many silos were occupied without revealing the status of any particular silo.

The winning contractor was TRW—Thompson-Ramo-Wooldridge. They worked with the National Security Agency to devise a scheme that was believed to solve this problem. Parts of it I won't address here, but the essential premise was there were a number of sensors which, if they could be emplaced in a silo, could reliably tell whether there was a missile in the silo or not. These were gravimetic sensors, tilt sensors, etc. Both parties accepted that there were sensors or combinations of sensors that could do this, but the problem was that the data acquired by these sensors (after all there is only one critical bit involved—"occupied" or "not occupied") had to be protected so that it couldn't be forged and couldn't be falsely attributed.

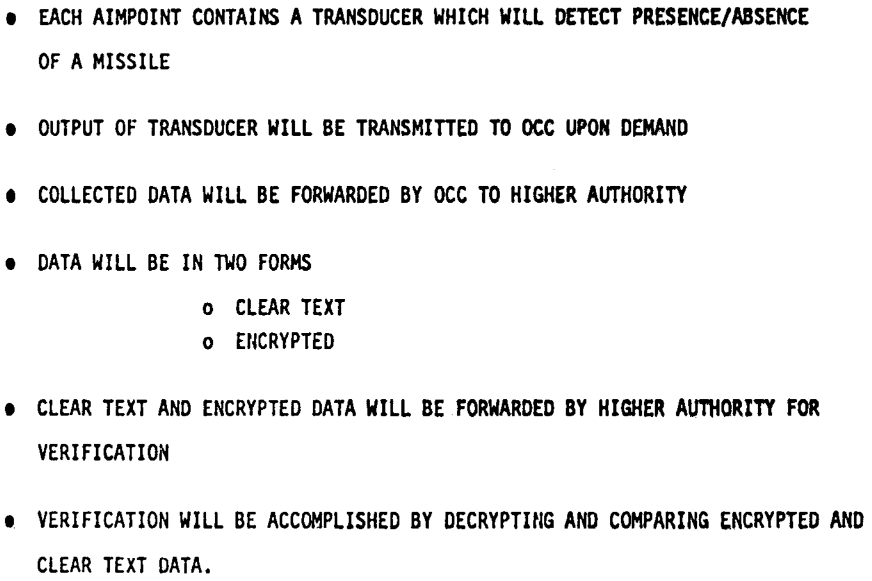
In other words the Russians should not be able to go to the United Nations and say "The Americans are cheating" and be able to present information that we couldn't disavow, showing that we were violating the terms of the treaty. Similarly, the United States should not be able to generate information that would deceive the Russians into believing silos are empty when they are occupied, etc. There are a long list of requirements which I won't recite in their entirety here. In the paper that I mentioned [1], the complete list of requirements for all of the parties is given. An obvious one is that neither side should be able to forge messages that would be accepted as authentic. The Russians might wish to, so they could falsely accuse the United States of cheating. We certainly would wish to be able to, so that we could field more missiles than we had to account for.

Requests had to be timely, otherwise we could merely interrogate the silo when there was no missile in it and save the response until the Russians issued a challenge, and then give them one saying the silo was empty when in fact it was occupied. An important portion of the anticipated treaty was that there would only be a limited number of challenges allowed each party, so that the Soviet Union couldn't say every day that they wanted to get a report on the Minuteman missile field. Hence, it was also important that we could only cause the transducers to respond when the Russians requested it, so that we couldn't exhaust their stock of challenges, when they hadn't issued them. I refer you to a paper of mine that appeared in *European Transactions on Telecommunications* for a complete discussion of the various competed needs of all of the parties [1].

Now Whit (Diffie) will be surprised to learn, since he knows that I can't remember anything and that I've thrown away all items of historical interest in my personal files, that I found a critical set of vugraphs from that period describing the proposed solution that resulted. These are briefing charts (Figs. 1–5) prepared by the TRW project manager, for a briefing to his upper management reporting on a briefing he'd given at NSA. They describe in some detail the TRW scheme and obviously are referring to a briefing TRW had just made to NSA. I don't remember his name since I only met him once when he came to Sandia to brief us—for reasons I'll explain momentarily—on the TRW study. This is part of my failing memory: I was lucky to find the vugraphs he gave us after the briefing!

What I discovered when I first saw the TRW study is historically interesting. Furthermore, it's going to be fun to describe, since it allows me to pillory the National Security Agency, one of my favorite pastimes. There is an ex-NSA man in the audience today (Robert Morris), so he may take umbrage at this. I need to explain a couple of things here. NSA saw no difficulty with the crypto processing that I'm going to talk about, because it had all been developed jointly with them, but I should emphasize that these vugraphs were used by the TRW program manager in reporting back to his management. This line down here (mentioning that Sandia should be brought in), I need to explain (Fig. 5). It was suggested to TRW by NSA that Sandia be asked to look at the transducer package as an extension of the Sandia code storage study in which we had developed very secure tamper proof and/or tamper sensing container technology.

We weren't asked to look at anything having to do with the crypto. That would have been unlikely then or now. The code storage study was a Sandia program to secure the enabling information for nuclear weapons in tamper resistant containers, the idea being that even though someone had unauthorized possession of the container it should be essentially impossible that they could get at the information inside. Sandia was also asked to supply some of the transducers, such as an incredibly sensitive motion sensor we had developed for a nuclear weapons application to make it impossible to undetectably



- EACH AIMPOINT CONTAINS A TRANSDUCER WHICH WILL DETECT PRESENCE/ABSENCE OF A MISSILE
- OUTPUT OF TRANSDUCER WILL BE TRANSMITTED TO OCC UPON DEMAND
- COLLECTED DATA WILL BE FORWARDED BY OCC TO HIGHER AUTHORITY
- DATA WILL BE IN TWO FORMS
  - o CLEAR TEXT
  - o ENCRYPTED
- CLEAR TEXT AND ENCRYPTED DATA WILL BE FORWARDED BY HIGHER AUTHORITY FOR VERIFICATION
- VERIFICATION WILL BE ACCOMPLISHED BY DECRYPTING AND COMPARING ENCRYPTED AND CLEAR TEXT DATA.

Fig. 1.

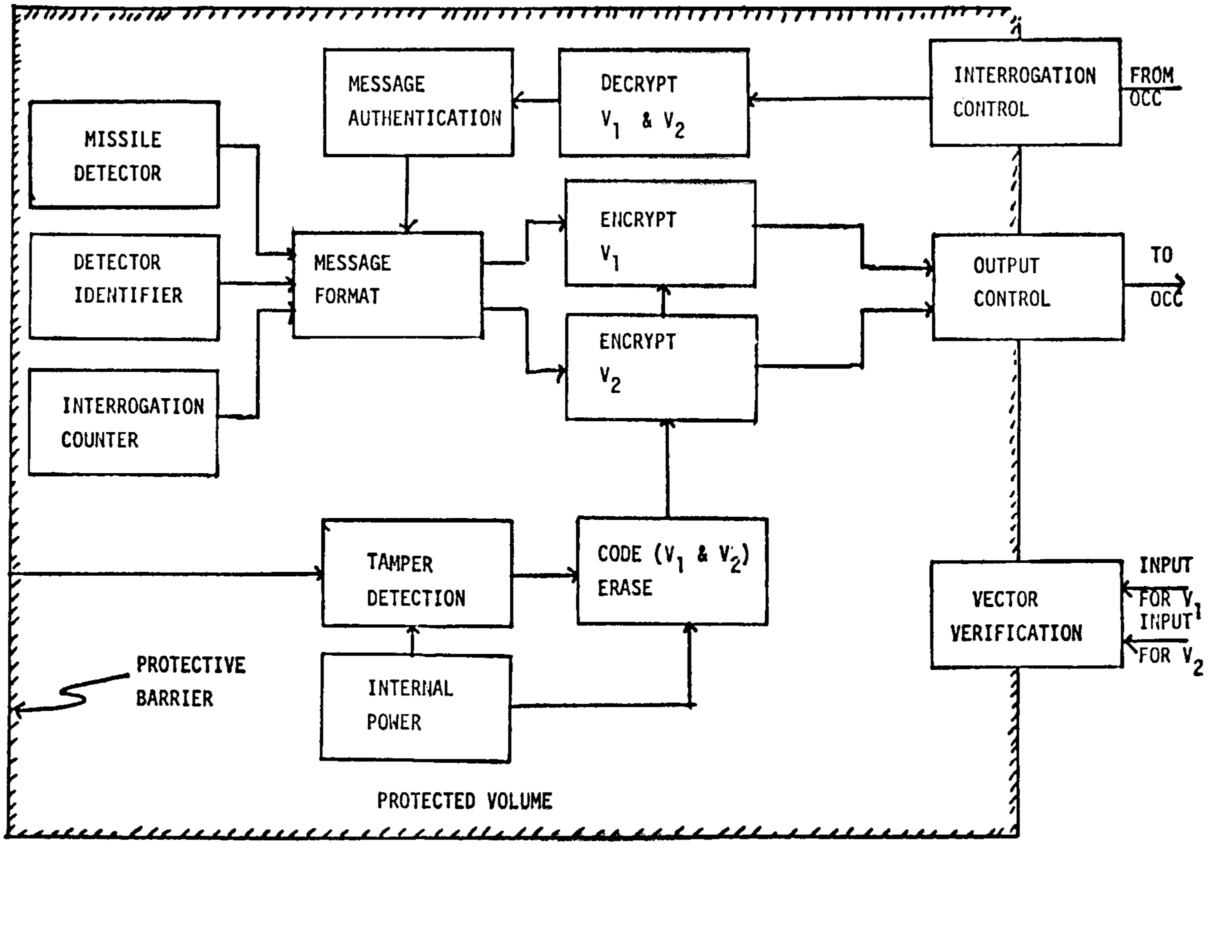
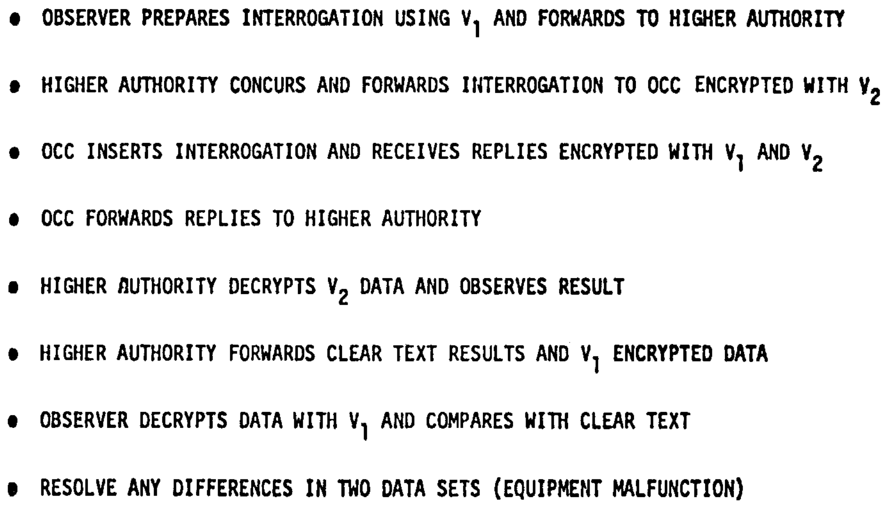


Fig. 2.

move the transducer packages after they were activated. These were to be active transducer containers, and so what they had in mind was that Sandia would apply this technology to protect the transducer package: The collection of instruments that could tell if there was a missile in the silo or not and also that a response had been generated by the equipment in a particular container. I'll go through what the response consisted of later.

Sandia was asked to come in on the program, sort of as an afterthought, and in a peripheral—although important—way. Everything else was considered to have already been settled. The crypto scheme that addresses the problem, and I'll talk about that in a moment, was all resolved—NSA had done that jointly with TRW and was fully satisfied with the scheme and the protocol. The transducers that would sense the presence of a missile either existed or else it was clear that they could be



- OBSERVER PREPARES INTERROGATION USING V_1 AND FORWARDS TO HIGHER AUTHORITY
- HIGHER AUTHORITY CONCURS AND FORWARDS INTERROGATION TO OCC ENCRYPTED WITH V_2
- OCC INSERTS INTERROGATION AND RECEIVES REPLIES ENCRYPTED WITH V_1 AND V_2
- OCC FORWARDS REPLIES TO HIGHER AUTHORITY
- HIGHER AUTHORITY DECRYPTS V_2 DATA AND OBSERVES RESULT
- HIGHER AUTHORITY FORWARDS CLEAR TEXT RESULTS AND V_1 ENCRYPTED DATA
- OBSERVER DECRYPTS DATA WITH V_1 AND COMPARES WITH CLEAR TEXT
- RESOLVE ANY DIFFERENCES IN TWO DATA SETS (EQUIPMENT MALFUNCTION)

Fig. 3.

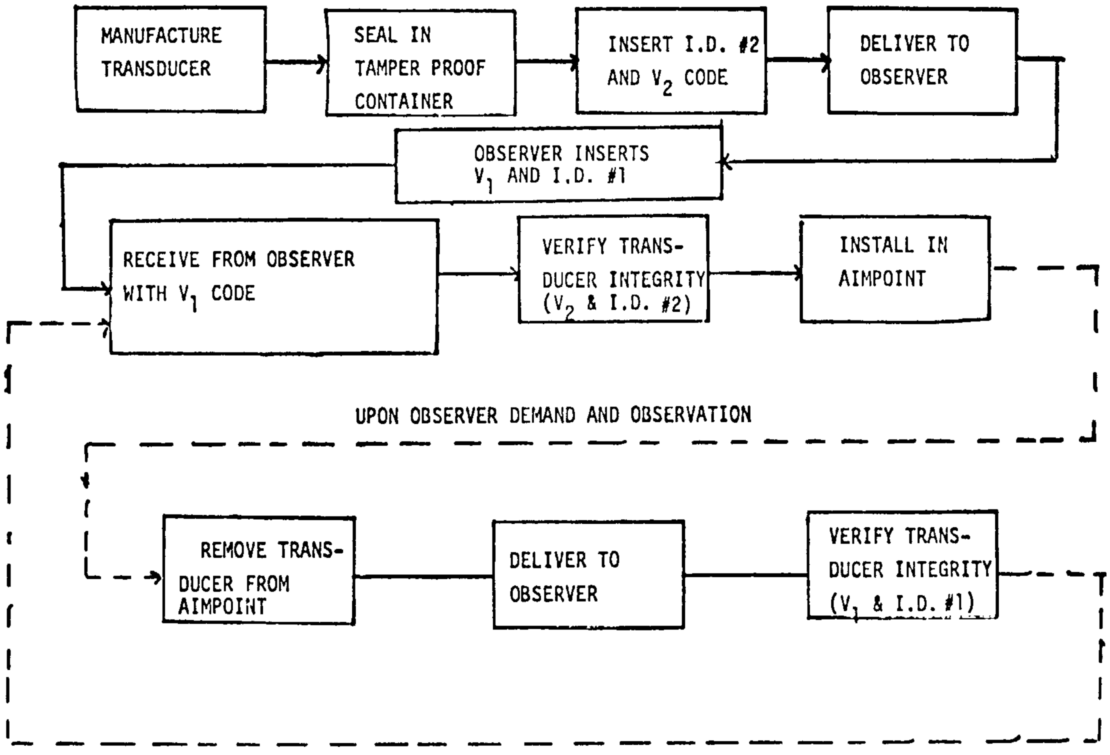
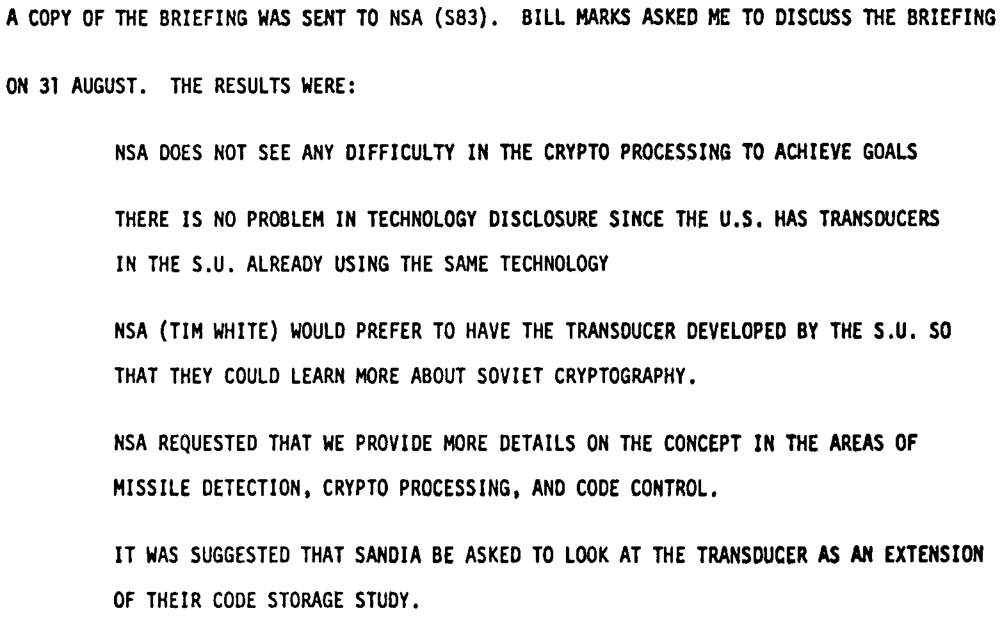

MANUFACTURE TRANSDUCER
SEAL IN TAMPER PROOF CONTAINER
INSERT I.D. #2 AND V2 CODE
DELIVER TO OBSERVER
OBSERVER INSERTS V1 AND I.D. #1
RECEIVE FROM OBSERVER WITH V1 CODE
VERIFY TRANS-DUCER INTEGRITY (V2 & I.D. #2)
INSTALL IN AIMPOINT
UPON OBSERVER DEMAND AND OBSERVATION
REMOVE TRANS-DUCER FROM AIMPOINT
DELIVER TO OBSERVER
VERIFY TRANS-DUCER INTEGRITY (V1 & I.D. #1)

Fig. 4.

developed. It was essential to have a believable secure store for these transducers and sensors that would prevent the package from being undetectably moved or tampered with which is why Sandia was approached by TRW.

The essential notion that NSA had endorsed—and for a long time I gave them a bad time for what I'm about to show you, without realizing there was a logical explanation of how that happened—was that they were going to use concatenated encryption as the essential element in solving the problem. The Soviet Union and the United States would each have a crypto algorithm (Fig. 3). At that time it was fashionable to talk about the key as keying variable, and so Vi merely represents the keying variable for the one party or the other. The purpose of the concatenation was to cause a cipher to be generated that couldn't have been generated by either party acting alone after the container was sealed. These Vi are secret; and are used in a sequence as shown in the block diagram. The resulting cipher could only have been generated by virtue of the source text having been operated on by the encryption systems with the keying variables of each party. Consequently, neither party alone could generate a fraudulent cipher. I don't even remember all the acronyms shown in the vugraphs. OCC is the Control Center but I don't remember what O stands for, perhaps Operational.

The essential notion that was supposed to protect the integrity of this information, was that the cipher that contained



A COPY OF THE BRIEFING WAS SENT TO NSA (S83). BILL MARKS ASKED ME TO DISCUSS THE BRIEFING ON 31 AUGUST. THE RESULTS WERE:

NSA DOES NOT SEE ANY DIFFICULTY IN THE CRYPTO PROCESSING TO ACHIEVE GOALS

THERE IS NO PROBLEM IN TECHNOLOGY DISCLOSURE SINCE THE U.S. HAS TRANSDUCERS IN THE S.U. ALREADY USING THE SAME TECHNOLOGY

NSA (TIM WHITE) WOULD PREFER TO HAVE THE TRANSDUCER DEVELOPED BY THE S.U. SO THAT THEY COULD LEARN MORE ABOUT SOVIET CRYPTOGRAPHY.

NSA REQUESTED THAT WE PROVIDE MORE DETAILS ON THE CONCEPT IN THE AREAS OF MISSILE DETECTION, CRYPTO PROCESSING, AND CODE CONTROL.

IT WAS SUGGESTED THAT SANDIA BE ASKED TO LOOK AT THE TRANSDUCER AS AN EXTENSION OF THEIR CODE STORAGE STUDY.

Fig. 5.

the information to be reported back to the Soviet Union could not have been forged by the United States because we couldn't create the Russian ciphers, nor could it be forged by the Russians to be falsely attributed to the United States since they couldn't create the U.S. cipher. The most important notion of all in this scheme was that both the clear text (and I'll come back to what that consisted of) along with the encrypted information would be output by the transducer package.

The final test was to compare the plaintext that had been sent out in clear with the result obtained by undoing the two encryptions. The United States would make this comparison as a test to decide whether to forward the message to the Russians or not; the Russians would make it to determine the authenticity of what they received. To repeat, you generate a cipher that can only be produced by encrypting with crypto variables that belong to the two nations, so that neither party can forge it, and you also pass out the plaintext that has been superencrypted to produce the cipher. As shown in the figure, both parties then decide whether the data is legitimate or not by decrypting it first with the one and then the other crypto scheme, in the appropriate order, and comparing the result with the plaintext.

Now comes the part wherein the problem lies. The bottom line in the figure indicating NSA's approval of the scheme is the item that is really inexplicable (Fig. 5). By the way, I should point out that ciphers from different silos had to be distinct, otherwise the United States could just interrogate an empty silo 1000 times. So each silo had to produce a silo unique signal, but the Russians couldn't know anything that would associate the unique portion of the plaintext with the silo it came from. The silo unique information consisted of a randomly generated identifier to be entered by the United States after the transducer packages were emplaced, and the Russians (and presumably the UN observers) had verified their correct operation. Obviously, the storage had to be write only, so that once a number was entered, it couldn't be modified without opening the secure data store. The Russians would have had to be satisfied that this was true before they would have accepted the scheme.

The point was that each silo would have a unique—but fixed—identifier associated with the output from its transducer package, but the Russians should not be able to associate the numbers with silo identities. It's just that they expected to see those serial numbers every time they received a response to a challenge they issued. The outputs also had to be in response to a query made by the Russians that could only be decrypted by their equipment—otherwise the United States could cause the transducers to spit out ciphers when the Russians hadn't asked for them.

But this bottom line in the figure is the crucial thing. This is the fatally flawed notion the NSA had endorsed. If when a cipher was decrypted, the decryption matched the plain text—and I emphasize, every bit of the plaintext was known in advance to the United States—the assumption was that the message (the complete text that was going to be passed along to the Russians) couldn't have even a single bit that was in equivocation to the United States. The United States knew the serial number of the unit; they knew the time; they certainly knew whether the silo was occupied or unoccupied. There was one bit of information there that was unknown in advance to the Russians. That was whether the silo was occupied or unoccupied. NSA's belief, that they acted on and accepted, was that if when you decrypted the cipher and the information was bit for bit identical to what it was supposed to be, then there could be nothing concealed in the cipher. We (Sandia) weren't invited to consider any of this. We were only brought in to make the tamper sensitive container for the transducers.

From the presentations you've heard today, you already know that NSA's assumption was not only wrong, it was fatally flawed in this case. I remind you that we're talking about 1976/77—the notion of public key cryptography had only appeared the previous year. We only had a couple examples of asymmetric cryptoschemes at the time, so it's hard for us now to think back to the cryptographic framework we were working in at that time. The NSA participants mentioned in the TRW vugraphs, Bill Marks and Tim White, both figured heavily here. Rick Proto, who shows up in my narrative in a moment, is currently head of the R (Research) Division of the NSA. These were heavyweights at NSA that were involved in both the evaluation of this scheme and in its approval.

As I've already pointed out, this was a time before we knew a great deal about public key cryptography. The people at the Agency, Tim White in particular as shown in the vugraph, were anxious that the Russians propose their own crypto scheme. As he said, it would reveal something about their crypto technology (Fig. 5). So it was proposed that the Russians be left to provide their own crypto system. The United States would put forward a crypto algorithm that we felt was acceptably secure, but not too revealing. We were going to ask the Russians to provide their own algorithm hoping it would give us a window into their technology. This is an important point to my narrative, since it left open the possibility that the Russians could have devised a crypto scheme of the sort I will describe in a moment. There were some general requirements, such as the output of the one algorithm had to be compatible as an input to the other etc., but in principle the algorithms could be totally different from each other.

Now why would the NSA, with their expertise in this area, make such an assumption? Well it's explainable in several respects. The first one is that you would expect this if you were conditioned to think in terms of classical (read symmetric) cryptography where you do not normally have two ciphers that decrypt to the same text with the same key. In other words, what you expect in symmetric cryptography is that if you take a text and encrypt it with two different keys, you get two different ciphers and these ciphers—if it's a good system—are going to be essentially two random bit streams with respect to each other. You don't expect to find any—well if it's a good system you'd better not find any—structure surviving from input to output. If you take the two ciphers and decrypt them with the two keys, you will get back the text.

I immediately spotted that it was possible (in fact I had an example in hand at this point) to devise a crypto system that has the following property. I'm going to do the 1-b example, but the concept is the same for any number of bits of covert communication. A crypto system that has the property that for every text and key pair, there are a pair of distinguishable ciphers that decrypt to that text with that key. When I say distinguishable, I don't care precisely what that means, say one cipher is odd and one is even, or one cipher is red and one is black—it doesn't matter. It's just that the ciphers in each of these pairs are distinguishable to me (with inside information), i.e., are different, and most importantly, that I can recognize the difference. Even though the ciphers are different, when they are decrypted with the common key one gets the same text.

Now if I—speaking as the Soviet Union—can get the United States to accept a scheme like that, then I can sneak through one bit of subliminal communication. In the TRW scheme, the United States first decrypts the cipher and compares the text with the text they know is supposed to be there. If it is bit for bit precisely what it is supposed to be (in the protocol that was approved by the NSA) the United States says that's fine, there's nothing concealed, and forward the plaintext/ciphertext pair to the Soviet Union as a response to their challenge. But consider what the Soviet Union does when they receive the cipher. Before they decrypt it, they look to see if it's the odd or even cipher—or whether it's the red or the black cipher, i.e., they look to see which cipher they've received. That gives them the subliminal bit of information. They then they go ahead and decrypt the cipher to recover the plaintext.

To complete this simple example, I'll reduce the Minuteman problem to where there are only two silos and a single missile in one of them. One silo's crypto equipment sends only the odd cipher while the other sends only the even. As a member of the emplacement team I know which equipment is where. What I don't know (initially) is the unique identifier number assigned to the transducer package by the United States after it was emplaced and turned on. This simple example would probably be too obvious to sneak by a vigilant host, but let's assume for the moment that the Russians were able to do this. The plaintext you get once you decrypt the message will tell you whether a silo is occupied or unoccupied, while the subliminal bit will identify which silo the response came from. So the one bit that came through subliminally would have completely defeated the purpose of shuffling the one missile around between the two silos. Obviously, after the first response, we would also have unambiguously identified the silos with their unique identifiers as well. Ten subliminal bits would have done the same thing for 1024 silos, i.e., for the Minuteman scheme that was being proposed by the Carter administration.

I thought this discovery so important, that I called for a meeting in 1978 at the NSA to tell them about the problem with the TRW scheme, and to describe subliminal channels as I then understood them. The meeting was held at the NSA facility at the Baltimore Friendship Airport, and was well attended. Bill Marks, Tim White, Rick Proto, and Brian Snow, all figures that we know, were present. Dick Leibler was there too, and much to his credit, was the only NSA person—then or later—to recognize the significance of what I was reporting. I made a thorough presentation on the topic, including a way to realize a 1-b subliminal channel.

The NSA response was, "Well, that was interesting, but there aren't any ciphers like that." Well there were: Even at that point in time. There was a convincing (to me at least) example that could be constructed using a result that had just appeared in public key cryptography. Since it hasn't been used, some of you may not have encountered the Rabin variant to the RSA scheme. Rabin's scheme was an early implementation of RSA: Its advantage being that on the one hand, you're essentially squaring to encrypt, and on the other hand you only have a $\log(n)$ difficult computation to extract a modular square root to decrypt. The encryption consists of taking the message m and forming $c = m*(m+b) \pmod{n}$ (b being a binary vector). Decrypting can't be done as it is in RSA though (by raising the cipher c to an exponent that is the multiplicative inverse to the encryption exponent with respect to the Euler ϕ function of the modulus), since 2 is not relatively prime to the Euler ϕ function of n, and hence has no multiplicative inverse. It is easy (order $\log(n)$ computational difficulty) to calculate the four values of m that would encrypt to c though.

It was this nonuniqueness of the inverse operation that was the weakness of Rabin's scheme for crypto purposes. A cipher formed by computing $m*(m+b)$ and reducing it with respect to the modulus n could be decrypted by essentially taking a square root, as you can easily see by completing the square on the right-hand side of the encryption expression. This is only $\log(n)$ difficult if you know the factorization of n, but provably just as difficult as factoring n if you don't. If m wasn't divisible by either p or q the completed square would have four square roots—which will be the case with probability $(1 - 1/p - 1/q)$, i.e., with virtual certainty.

What that means is, for a given cipher, there are four choices for the message. Later, Hugh Williams came up with a scheme that got rid of this ambiguity by requiring p and q to come from specified residue classes with respect to eight which allowed him to specify a canonical message out of the set of four square roots. This isn't important to my narrative though, since I saw how to adapt the ambiguous Rabin scheme to provide a one bit subliminal channel. This was the example that was sitting there, waiting to be applied.

What we were concerned with was not secrecy, since the plaintext was sent in the open, but authentication or signatures. The important thing in the scheme that I've described, was that the United States and the Soviet Union would each sign the cipher, by carrying out an operation that each of them was uniquely able to do, but which the other could verify. Both parties were to be able to decrypt the ciphers unilaterally, but it took both of them—or to be precise, their proxy crypto systems inside the tamper sensing container—to generate a cipher. The crucial thing was that the message was to be authenticated or signed. This could be done by using the inverse of the Rabin scheme. In other words, instead of encrypting by using $m * (m + b) \pmod{n}$ when we had a message, we extract a square root of the completed square of the encryption expression (of which there are four in general) and one of those is the "cipher" that is sent. Verification of a signature required only that the cipher be squared mod n and the result compared to the plaintext.

Now we have a crypto scheme that has the property that for every message there are four ciphers, all four of which decrypt to the same message with the same key. Furthermore the four ciphers belong to four classes that are easy to recognize. How do I recognize them? Well the Russians knew the factorization of their modulus so it was an easy task for them ($\log(p)$ plus $\log(q)$ difficult) to calculate the Legendre symbol of the message with respect to each of the primes. That is to say, they looked to see if the root they received was a quadratic residue with respect to each of the two primes. So it was trivial for them, when they received a cipher, to put the cipher into one of four classes, which should be good for two bits of subliminal communication.

As I think Rick Proto said after my presentation, "We'd never accept a system like that," and it's true that the United States could limit the Russians in this example to one subliminal bit. This is because the United States could calculate the Jacobi symbol of the cipher without knowing the factorization of n and could insist that the only ciphers they would forward would have a specific Jacobi symbol of either $+\,1$ or 1. Within one of these classes, however, it is impossible to distinguish the two members without knowing the factorization of the modulus n. If the Jacobi symbol was a –1 the United States would know that the cipher was not a quadratic residue with respect to one of the primes, but they would not be able to say which one. Similarly, if the Jacobi symbol was $+\,1$, they would know that the cipher was either a quadratic residue with respect to both p and q, or else both were quadratic nonresidues, but they couldn't distinguish between the two cases. The result is that it is not possible to close the one bit subliminal channel in this example. This remains one of the staple items in all sorts of protocols for us today.

Again, NSA's response was, "That's silly, we wouldn't accept a system like that." Furthermore, a 1-b existence proof wasn't regarded as a real threat. Ten bits were needed to identify the silos in the Minuteman concealment scheme, and this was only a 1-b channel. A 1-b existence proof wasn't enough to convince them. As a matter of fact, it was a more serious threat, even at that point, than I realized.

Remember, the elliptic curve factoring technique was a long way in the future. The best factoring method we had in those days, was either CFRAC (the continued fraction algorithm), or the quadratic sieve. We didn't even have the present powerful versions of the quadratic sieve. It was to be some time yet before Davis and I and subsequently Peter Montgomery would develop the techniques that made the quadratic sieve so powerful. But the point I wanted to make about those factoring techniques is that they could not distinguish between numbers of special form and numbers of general form. So at that time if I had a number whose factorization I desired, say a number that was 200 decimal digits in size, and it was made up of five roughly 40-digit components, the fact that there were 40-digit factors didn't aid me in factoring at all. The only choice was to run one of the general purpose factoring routines.

Now I need to remind you what the state of the art of factoring was at that time. 1978 is the year that the Sandia Labs fielded the first implementation ever made of RSA. This was for controlling access to the zero power plutonium reactor at Idaho Falls. Very few things were more sensitive. Only a nuclear weapon perhaps is more sensitive, because the very character of a plutonium pulse reactor is that you have a supercritical mass of plutonium. It's bare and you bring it together; in other words, you bring the assembly right to the point of nuclear explosion, and study the onset of the nuclear reaction. It isn't even in pieces, nor is it concealed. It is a mechanism that has more plutonium than you need to make a bomb. If you brought the two pieces too close together you'd have not a bomb, but a disastrous reaction. An accident of this sort happened with fatal consequence at Los Alamos several years ago.

So you want to have carefully controlled access to a zero power plutonium reactor. We implemented an RSA-controlled portal into the place where there was access to this plutonium.

The point of my story is this: we gave a lot of consideration at Sandia as to how large a modulus was needed. Note that we are not talking about cryptographic keys, which may have critical value even if they are stale—witness the Walker spy case—but rather access control, whose only value is contemporary. We wanted the modulus to be large enough, so that the difficulty of factoring it would define a suitable level of security for the reactor, but we also wanted to not make it larger than necessary so as to not make the computational burden greater than necessary.

We were caught between a rock and a hard place. In 1978, 334 b, roughly 100 decimal digits, was orders of magnitude beyond anyone's ability to factor. We implemented the access control for the plutonium reactor using RSA and a 334-b modulus. This was done using discrete components. It wasn't just that we didn't have special purpose circuits available—TRW had special purpose 16 × 16 b multipliers in a single chip implementation (that ran red hot) but it was easier to design with general purpose logic chips, than to make a Rube Goldberg design around a few special purpose chips. VLSI wasn't that far along yet, and so you were compelled to do no more computation than you had to. By that I mean only the amount that you had to do to be secure, so no one would have suggested using a 200-, 300-, or 400-digit modulus. Moduli of this size, and the need for the security they provide, were to come much much later.

But let's go back now to the scenario that I was describing a moment ago using the Rabin's variation on RSA in the number theoretic setting just described. Since we couldn't take advantage of a comparatively small factor in the modulus to peel it off, had we gone to 160-digit modulus, we could have easily made it up of four 40-digit primes which would have been far beyond anyone's ability to factor in those days. A number of this size—160 digits—with no small factors is still moderately difficult to factor, but using the elliptic curve factorization technique, it is now easy to peel off the 40-digit factors. As I said though, the elliptic curve technique that exploits smaller factors was still a long time in the future, and so we could have concealed a number of bits in a Rabin type signature using such a modulus because the number of square roots grows as a power of 2—2 to the power of the number of factors.

Consequently, the existence proof 1-b subliminal channel I presented in 1978 was already a threat that wasn't taken seriously. In other words, using what I have just described—in a subliminal channel which to the best of my knowledge the NSA could not have detected at the time—the uncertainty to the Russians in the Minuteman concealment scheme could have been cut by a factor of 16, which probably was already enough to defeat the purpose of the missile shell game. Fortunately that wasn't what the decision to abandon the missile shell game hinged on. It was the silliness (and cost) of it all—shuffling these 100 missiles round amongst 1000 holes in the ground—that ultimately killed it. But this was the origin of the subliminal channel.

Since I'm covering history, I want to talk about the problem of constructing two or more ciphers that decrypt to the same meaningful text with a single key. I said earlier that it is hard to find ciphers of this sort. If it's a good cipher, you expect that going the other way round, i.e., that given a plaintext encrypting it with two different keys should give two uncorrelated, i.e., apparently random, numbers.

The next two examples are more for David's (Kahn) pleasure than anyone else's. This is just fun and games—but it does illustrate a serious point. This is a cipher I constructed specifically for this example which we are going to decrypt by simple (schoolboy) substitution. The setting is that this cipher has been intercepted on a Persian courier in the time of the Greek and Persian wars.

EQDGZWMWLMHNWPQVFMWN

Ciphertext:	ABCDEFGHIJKLMNOPQRSTUVWXYZ	
Plaintext:	FGJBMAIHKPQNTDSCOUVWRXEYZL	(KEY)

MOBILE TENTH DECORATED

If we use the substitution key shown first, this is the text that pops out. It is a meaningful text. I don't know that they had that many foot armies in the field in Greece, but when we decrypt the cipher, it says "mobile tenth decorated." However, the same cipher, when decrypted with a different key gives a totally different text—with a much different meaning:

EQDGZWMWLMHNWPQVFMWN

Ciphertext:	ABCDEFGHIJKLMNOPQRSTUVWXYZ	
Plaintext:	MWFGAPISKBZXTDHCRNOLJYEQUV	(KEY)

ARGIVE TEXTS DECRYPTED

Well I made a couple of such ciphers. By the way, these aren't easy to make because even substitution ciphers are kind of random and unicity distance catches up with you at around 25 to 30 letters. It isn't that easy to make up a cipher that decrypts into two meaningful texts, but it becomes especially difficult if you place constrains on the text you are willing to accept—as I have done in these two examples. With the "mobile tenth decorated," you will observe that the word breaks and word sizes have been preserved between the ciphertext and the plaintext.

In the second example I've taken greater liberties with the construction—that was because I was having trouble making another example that had a sort of cryptographic content to one of the plaintexts and still had the desired property. I'll ask you, since we're only playing, to give me the freedom to put the wordbreaks where I want them.

QWNMFOSRWHWRPGFOWQWN

Ciphertext:	ABCDEFGHIJKLMNOPQRSTUVWXYZ	
Plaintext:	BDFGHPXSJKMOYTLECNARUVIWQZ	(KEY)

CITY PLAN IS INEXPLICIT

This is the ciphertext, no particular setting for it, which when decrypted with the first substitution key, says, "City plan is inexplicit." I have no idea what the setting might be for that text, but it is arguably a meaningful plaintext. However, with the other substitution key it decrypts to "(the) red t(ele)phones (are) enciphered":

```
            QWNMFOSRWHWRPGFOWQWN

Ciphertext: ABCDEFGHIJKLMNOPQRSTUVWXYZ
Plaintext:  MWFGAPISKBZXTDHCRNOLJYEQUV   (KEY)

            RED TPHONES ENCIPHERED
```

One of my reasons for showing you these examples was to illustrate one of the reasons NSA was so confident that if the plain text matched the decryption of the cipher text bit for bit, there couldn't be anything concealed. It is precisely because of the difficulty of doing what I've just shown you. These were toy examples and even so they are still difficult to construct. But the NSA had an even better reason to believe as they did, and I didn't tumble on this until much later. At the time I'd been working at Sandia—that sounds a little too pretentious so let me change that to say—Sandia had been working with the Department of Defense and the Arms Control and Disarmament Agency with assistance from the NSA for ten years, developing unattended seismic observatories to monitor compliance with a contemplated comprehensive nuclear weapons test ban treaty. We'd had treaties dating back to 1962 limited testing of nuclear weapons in the atmosphere, in the oceans and in near space, but both sides continued until recently testing nuclear weapons underground, and one of the reasons that loophole was left was the because of the difficulty of verifying compliance.

Each party could verify by national means—in this case by seismic nets (of which we have one stretching up through Scandinavia still) whether the other side had tested nuclear weapons down to a threshold on the order of 100 kilotons at these distances. Now what was desired was to negotiate a treaty in which the detectable threshold (so that you could tell if people were cheating) dropped down to the order of one kiloton, because it was believed that no meaningful weapons development, and hence advancement of the state of art in nuclear weaponry to gain an advantage over the other side, was possible if you couldn't actually field test devices of greater than one kiloton.

What we (the United States) had done was to develop families of seismic sensors and the algorithms and technology to analyze that data, which would allow us, if we could get in a little closer with unmanned sensors, to tell if the other side was testing, and hence cheating. This technology had been under active development for some time and those of you that have a copy of the IEEE book on *Contemporary Cryptology*, the last chapter in there is devoted to how to ensure that data taken to verify compliance with such a treaty is trustworthy [2]. Now I've made a lot of fun of NSA thus far, so now I'll make a little fun of myself. Over a period of roughly a decade, I kept thinking I had completely solved this problem, only to find a new facet to the problem that I had completely overlooked before.

At first it was crucial that the data be authenticated; obviously if you can't trust the data, there's no point in putting the sensors out there. Then it was essential that the data be verifiable to third parties because we realized that if we caught the Russians cheating, or vice versa, then the aggrieved party would almost certainly go to some third party—the United Nations, NATO, or the world community and say, "The other side is violating the treaty and here's the proof." And so little by little, we saw success in steps. At the end of the chapter I conclude by saying a number of things. No part of the message can be concealed, in particular from the host (that means the host nation who is allowing the other side to put sensors in their territory). I've already indicated at that time we didn't know about subliminal channels and so that statement isn't quite true.

But the relevance of this to NSA's assumption (that if the decryption of the ciphertext matched the plain text, nothing was concealed) was that ten years of intensive work had been devoted essentially to achieving this long list of functional abilities that were needed for treaty verification by concatenating encryption. For ten years we'd been developing a progressively more complicated scheme in which we had ensured that the interests of all the parties were protected by using concatenated encryption, so that you ended up with ciphers that no party or anticipated cabal of parties could have forged. So long as you didn't reveal your key, no one would ever be able to falsely attribute a message to you, etc.

In a sense, we (Sandia) set NSA up—there was no malicious intent in this, but they were conditioned to believe in concatenated encryption where you compared the plain text to the decryption of the ciphertext as a means of insuring the authenticity of data. So it isn't totally inexplicable that they would have made the assumption they did.

I will close my narrative rather quickly now. That's the history of how the subliminal channel came to be. We quickly began devising practical subliminal channels. By practical, I mean ones that were edging up to getting enough information through to be of real use; not just one bit, or two bits, or a few bits as we'd done initially, but a meaningful amount of information. And now I am going to repeat myself, and Ross (Anderson) and Yvo (Desmedt).

Ross spoke this morning about subliminal channels in El Gamal signatures and waved his hands a little about similar channels in signatures generated using the U.S. digital signature algorithm (DSA). I only want to repeat a couple of things; the digital signature algorithm, now the part of the digital signature standard, is as Ross described it [3]. You have a modulus whose size is between 512 and 1024 b in 64-b increments depending on the security you want. You choose a prime q of 160 b and that's where most of your signature security is going to reside. The exact steps or details I think are known to everyone here.

By the way, this would have been a fully acceptable scheme at the time the pair of treaties were being negotiated. NSA was planning to ask the Russians to put forward their scheme. So had this been put forward, we would have assuredly accepted it. The user chooses a secret key, the knowledge of which is equated with his identity, so he'd better protect that. If he lets that get away, he's letting his identity get away. That's x in the vugraph. He then does a modular exponentiation of the publicly known element g to produce a public version of his key.

Ross showed you this morning the system for generating the signature, so I won't repeat that since it isn't essential to the point I wish to make. When a message is to be signed, it is first hashed down to a standard size. The hashing generates an element in $\mathrm{GF}(q)$. The crucial point is that the signer next chooses a session key k, a "random" element in the field. If he's executing the protocol faithfully, he flips a 2^{160} sided coin and gets a random element. He then calculates these two quantities r and s, and the important thing is not how that's done, but rather that the signature to a message consists of two 160-b extensions. The final signed message is the concatenation of the original message and those two 160-b quantities. The security of the signature against forgery is just the probability given m and s say, of choosing an r that is a companion to them under the operations Ross described in detail this morning, that is to say one in 2^{160}.

So we have 320 b of equivocation in the signature, 160 of which are used for security, and the other 160 of which are potentially available for subliminal communication. There is no necessity that there be 320 b of redundant information to get 160 b worth of security, but in all cases you have some superfluous bits hanging out there which may be convertible to subliminal communication. The history of subliminal channels has been the recognition and exploitation of this fact.

At this point, something important with respect to subliminal signatures or communications needs to be said. There are two types of subliminal channels distinguished by whether the subliminal transmitter unconditionally trusts the subliminal receiver or not. In a scheme of the type just described which typifies many digital signature schemes—not all but many—if I wish to communicate subliminally, and I'm willing to unconditionally trust the intended subliminal receiver, meaning I'm willing to give him the ability to utter undetectable forgeries of my signature then, as Ross pointed out this morning, we can use the full equivocation of the remaining 160 b. As a matter of fact, it is in principle possible to use all the equivocation in a signature that isn't used for security to buy subliminal communication. Whether you can actually do this or not will depend on the particular signature scheme. In this case it is possible, but at the expense of having to unconditionally trust the subliminal receiver.

Well now in the scheme I was describing for resolving the dilemma between the two treaties, naturally the Soviet Union unconditionally trusted the receiver—since they were the receiver. Hence, the equipment in the transducer box could have transmitted, had they proposed the DSA for their signature algorithm, the full 160 b, while they only needed 10 to completely defeat the Minuteman concealment. But the point is, in that setting, the subliminal transmitter and the subliminal receiver are one and the same in the sense that they work with common purpose. Now there are many other instances or applications in which the subliminal transmitter isn't willing to unconditionally trust the receiver, and that leads to the second class of problems concerning subliminal channels. How much information can you get through when the subliminal transmitter considers the receiver suspect, i.e., he's either only willing to partially trust him or not at all.

My object is not to repeat what Ross said, but to illustrate another concept. Go back to the example I used at the time I tried to persuade the NSA that subliminal channels were a bona fide threat, in the setting in which I devised them. I want to use the same number theoretic principle to describe to you now a subliminal communication that appears very strong. Why was subliminal communication possible in signatures generated using either the El Gamal or the DSA? It was possible because the subliminal transmitter did not have to behave faithfully. The protocol assumed he was going to draw the session key k randomly, but he didn't have to do that. Furthermore, there was no way that an observer could tell from what range he chose the key or with what probability distribution. So the signer could deliberately choose the key to convey the information he wanted to send. That communication was totally dependent on the fact that the subliminal transmitter was free to pick the key k. If you want to deny him that ability you must take away from him the freedom to choose k.

Although it isn't obvious, no one else can choose, or even know, the session key either, since that information would make it possible for them to utter undetectable forgeries of the signers signature. Hence in order to close subliminal channels and maintain the integrity of digital signatures, no one individual can choose the session key. I have devised and reported an interactive protocol between two parties—the signer and a trusted key generation bureau (the KGB) that achieves this end.

But what I wanted to close by showing you, was a neat result harking back to the example where we were looking at quadratic residuosity with respect to the prime factors of a composite modulus. We will do something quite different here. The subliminal transmitter and its receiver choose a large prime known only to them. Their convention is going to be that when a signature is seen they will calculate the Legendre symbol of the signature with respect to this prime which only they know. They will get a binary bit $(+1$ or $-1)$. That is a fair procedure in the following sense: If we exclude the collection of numbers whose square is less than a given prime, the remaining collection of numbers less than the prime will have quadratic residues distributed 50/50.

So from the standpoint of an observer looking at the signature that's sent, the subliminal transmitter must be able to manipulate the signature by the randomness that he puts in, but all he's doing is causing the signature that he's willing to transmit forward, be in the appropriate quadratic residue class with respect to a particular prime, i.e., to have the right Legendre symbol with respect to this prime that only he and the receiver know. So this is completely fair and unbiased, and hence undetectable.

Now what if he wishes to send ten bits as was needed to defeat the Minuteman concealment scheme? Here number theory isn't quite up to our needs. There is every reason to believe that if we chose ten large primes at random, and chose residues less than these moduli, that the number of occurrences of each of the possible ten bit binary numbers as labels of the residue classes with respect to the primes would be uniform. We don't know of any instance in which the quadratic residue/nonresidue sequence for two primes are related, but we have no proof of that either. So all that I can prove is that this is a secure and sound channel for sending one bit. Whether it is an equally secure and sound channel for sending ten bits would depend on whether ten randomly chosen primes and randomly chosen residues, would uniformly map out all possible 2^{10} residue/nonresidue classes between them. It is almost certainly true but I have no idea of how to go about proving it.

I will close by returning to approximately where I started, and pointing out that the quadratic residue technique that provided the existence proof that demolished the assumption NSA had made—that if the decryption of the ciphertext matched the plaintext nothing could be hidden—also provides a technique that appears to offer the possibility of communicating subliminally so long as the transmitter has the freedom to accept or reject signatures, even if he can't force the choice of the session key.

Thank you for your patience and attention.

Question: How can the Soviets be sure that you didn't just put some noise source in that will generate a biased probability or some other hardware hack like that?

Simmons: The plan was that the Russians would build their own crypto-hardware and the United States would build theirs and the sensors would have been evaluated and accepted by both parties. A point I'll make here is that when they issued a challenge and asked for the status of the Minuteman field, the response to that query had to be a response from all 1000 sensor packages. If they failed to get a sensible response from any sensor package, then ipso facto the United States was violating the treaty. They could then go to the United Nations and say, "It looks like the Americans are playing fun and games with us."

Question: So if they did this with random messages, you would get approximately the right number of missiles and could get fluctuations.

Simmons: That would have been easy to detect. There is a long list of conditions the system had to satisfy—and there are surprisingly many—which I couldn't describe in the time available today, most of which had no bearing on the discovery of subliminal channels. It turns out that they can all be satisfied except that the subliminal channel which I've described to you is left open. If you are interested in a more complete description of the system, I would refer you to the paper that appeared in the *European Transactions on Telecommunications*.

Are there any other questions? Then I would like to thank you again for your attention.

REFERENCES

[1] G. J. Simmons, "Subliminal channels: Past and present," *Eur. Trans. Telecommun.*, vol. 5, no. 4, pp. 459–473, July/Aug. 1994.

[2] ——, "How to insure that data acquired to verify treaty compliance are trustworthy," *Contemporary Cryptology—The Science of Information Integrity.* New York: IEEE Press, 1992.

[3] R. J. Anderson, S. Vaudenay, B. Preneel, and K. Nyberg, "The Newton channel," in *Proc. 1st Int. Workshop on Information Hiding*, Cambridge, U.K., Springer Lecture Notes in Computer Science, vol. 1174, May/June, 1996, pp. 151–156.

Gustavus J. Simmons received the Ph.D. degree in mathematics from the University of New Mexico, Albuquerque, in 1969.

He retired in 1993 as a Senior Fellow and the Director for National Security Studies at the Sandia National Laboratories, Albuquerque, NM. Earlier he was Manager of the Applied Mathematics Department and Supervisor of one of the two divisions at Sandia devoted to the command and control of nuclear weapons. In all of these positions he was primarily concerned with questions of information integrity arising in national security: command and control of nuclear weapons, verification of compliance with various arms control treaties, individual identity verification at sensitive facilities, etc. His research has been primarily in combinatorics and graph theory and in the applied topics of information theory and cryptography, especially as applied to message authentication and systems design to achieve this function. At present his research is devoted primarily to the problem of devising protocols that can be trusted to function correctly, even though some of the inputs and/or participants may not be trustworthy, and of proof techniques for the integrity of such protocols.

Dr. Simmons was the recipient of the U.S. Government's E. O. Lawrence Award in 1986. In that same year, he also received the Department of Energy Weapons Recognition of Excellence Award for "Contributions to the Command and Control of Nuclear Weapons." He was awarded an honorary Doctorate of Technology in May 1991 by the University of Lund (Sweden) in recognition of his contributions to communications science and to the field of information integrity. In 1996 he was made an honorary Lifetime Fellow of the Institute of Combinatorics and its Applications. Dr. Simmons has published over 150 papers and books, many of which are devoted to the analysis and application of asymmetric encryption techniques or to message authentication. At the invitation of the editors, he wrote the section on cryptology that appears in the 16th edition of the *Encyclopedia Britannica*.