A BIT BY BIT SECURE PUBLIC-KEY CRYPTOSYSTEM

by

S. Goldwasser and S. Micali

Memorandum No. UCB/ERL M81/88

4 December 1981

A BIT BY BIT SECURE PUBLIC-KEY CRYPTOSYSTEM

by

Shafi Goldwasser and Silvio Micali

Memorandum No. UCB/ERL M81/88

4 December 1981

ELECTRONICS RESEARCH LABORATORY

College of Engineering
University of California, Berkeley
94720

A Bit by Bit Secure Public-Key Cryptosystem

Shafi Goldwasser and Silvio Micali***
University of California - Berkeley

Abstract

How to implement **a provably secure** Public Key Cryptosystem is the most challenging task in modern Cryptography. And sending messages consisting of a single bit in a secure way is certainly one of the most challenging problems in a Public Key Cryptosystem ! Without any special ability, an eavesdropper has a 50% chance of guessing 1-bit messages correctly. By sending 1 bit securely we mean that no eavesdropper is able to guess correctly whether the message is 0 or 1, 51 times out of 100.

No Public Key system currently exists for which we can prove that decoding 1-bit messages is computationally infeasible. Here computationally infeasible means equivalent to a problem such as factoring, index finding or deciding quadratic residuosity modulus composite numbers.

In this paper we present a way of sending 1-bit messages in a Public Key environment and prove that if an eavesdropper can guess these messages correctly $50 + \varepsilon$ times out of a 100, for some $\varepsilon > 0$, then he can decide quadratic residuosity modulus composite numbers in Random Polynomial Time.

Given a large composite number N, Rabin found a 4-to-1 function f which is as hard to invert as factoring. This result marks a great achievement in Cryptography. f can be used to build a Public Key Cryptosystem in which numbers chosen at *random* from $[1, N]$ can be encrypted in a way such that decoding is provably as hard as factoring. However, Public Key Cryptosystems are not generally used for sending random numbers between 1 and N, but to send **messages**.

We show that if M, the set of messages, is sparse in $[1, N]$ (e.g. let M be the ASCII representation of English sentences), then inverting f on M (i.e. decoding) is **not** provably as hard as factoring.

We also show how to overcome the above difficulty by providing a Public Key Encryption Function for sending messages belonging to a sparse set, for which we can **prove** that decoding is as hard as deciding quadratic residuosity modulus composite numbers.

This research was supported by
* NSF Grant MCS-79-03767 and
** fellowship from Consiglio Nazionale delle Ricerche - Italy and in part by the above mentioned grant.

1. IS IT REALLY DIFFICULT TO SEND A SINGLE BIT SECURELY IN A PUBLIC KEY CRYPTOSYSTEM ?

1.1 What is a Public Key Cryptosystem

The concept of a Public Key Cryptosystem has been introduced by Diffie and Hellman in their ingenious paper [4]. In short, let

M = finite message space,

$A, B,...$= users,

$m \in M$,

E_A = A's encryption function from M to M, which is ideally bijective, and D_A = A's decryption function such that $D_A(E_A (m))=m$ for all $m \in M$. In a Public Key Cryptosystem E_A is placed in a public file, and user A keeps D_A private. D_A should be difficult to compute knowing only E_A. Thus, to send message m to A, B takes E_A from the public file, computes $E_A(m)$ and sends this message to A. A easily computes $D_A(E_A(m))$ to obtain m.

Several implementations for a Public Key Cryptosystem have been proposed. Among them we would like to mention the one by Rivest, Shamir and Adleman, the RSA scheme [7], and its particularization suggested by Rabin [6]. In this latter scheme, Rabin produces user functions E_A which are as hard to invert, on a generic input, as factoring.

1.2 Attempts to send a single bit securely in a Public Key Scheme.

Assume that user B wants to send a single bit message to user A in great secrecy. B wants that no eavesdropper can have a 1% advantage in guessing correctly his message. B knows that E_A is hard to invert and tries to make use of this fact in the following way.

Attempt1: B selects $r \in M$ at random and sends $E_A(\mathrm{r})$ telling that his bit is the ith one in the decoded message (i.e. in r).

A can decode and thus get the desired bit. But what can an eavesdropper do ?

Danger: let $\mathrm{y} = E_A(\mathrm{x})$, where E_A is a one way function. Then, given y it could be difficult to compute x but not a **specific bit** of x.

Example: let p be a large prime such that p-1 has at least one large prime factor. Let g be a generator for Z_p (Z_p is cyclic if p is prime). Then g^x mod p is a well known one-way function. But, even though given g^x mod p it is difficult to compute x (the index finding problem), it is easy to get the last bit of x !

In fact, x ends in 0 if and only if y is a quadratic residue mod p (i.e. if the equation $\mathrm{y} = z^2$ mod p is solvable), and for p prime we have fast random polynomial time algorithms to test quadratic residuosity !

We just saw that given $\mathrm{y} = \mathrm{f(x)}$, for some one-way function f, some particular bits of x are totally insecure. It could also be that, given $\mathrm{y} = E_A(\mathrm{x})$, an eavesdropper is able to guess correctly any bit of x with probability 60% and still is not able to find x ! Thus it is easy to see that the following attempt (suggested by Donald Johnson) to send a single bit in a Public Key System is also dangerous.

Attempt2: B sends $E_A(\mathrm{x})$ to A, where x is, say, 100 digits long. The first 50 digits of x specify a location i (i=50,...,100). The bit B wants to send is the ith bit of x.

The following kind of attempt may help in clarifying the difference between

Private Key and Public Key communications.

In [3], Blum and Micali also show how two partners A and B can exchange single bits securely if they share the knowledge of a secret integer s chosen at random in a big interval. Assuming that index finding is hard, they prove that an eavesdropper, if he has no idea about s, cannot have a 1% advantage in guessing whether a given message m means 0 or 1. The following attempt was suggested by Andrew Yao.

Attempt3: As s needs to be chosen at random, B could send it securely to A in a Public Key Cryptosystem by sending E_A(s), where E_A could be the Rabin's function. Then B sends a message m like in the Blum-Micali scheme. A, who now knows the secret s, correctly interprets it; but an eavesdropper cannot have any advantage in guessing it.

The problem with this is that Blum and Micali show that no eavesdropper can have an advantage in guessing m only if s is **totally unknown to him**. But the knowledge of E_A(s), could unable the eavesdropper to get a slight advantage in guessing the meaning of m! In fact any information that the eavesdropper can get out of E_A(s) (not enough to invert E_A of course !) could help him in doing better than guessing at random.

Conclusions

There are infinitely many ways in which a single bit could be "embedded" in a binary number x. Taking the "exclusive or" of all the digits of x is just one more example.

Given y =E_A(x), being able to find out one bit embedded in x **does not con-**

tradict the fact that it is hard to get x.

If we do not know, as it is true for the current status of the research, which bits embedded in x are easy to discover, then **what is a secure way to send a single bit ?**

2. A RESULT IN NUMBER THEORY

Let $Z_N^* = \{ x \mid 1 \le x \le N-1$ and x and N are relatively prime $\}$.

2.1 Background

Given $q \in Z_N^*$, is q $=x^2$ mod N solvable ? If N is prime, then there is an easily computed condition for the solvability of q $=x^2$ mod N; if a solution exists, q is said to be a **quadratic residue mod N.** Otherwise q is said to be a **quadratic non residue mod N.** From now on let p_1, p_2 be odd primes and $N = p_1 p_2$. Then q $=x^2$ mod N is solvable if and only if both q $=x^2$ mod p_1 and q $=x^2$ mod p_2 are solvable. If this is the case, q is said to be a **quadratic residue mod N** , otherwise q is said to be a **quadratic non residue mod N.**

The Jacobi symbol (q/N) is so defined: (q/N) = (q/p_1) * (q/p_2) , where for all $x \in Z_p$, p odd prime, (x/p) = +1 if x is a quadratic residue mod p and -1 otherwise.

Despite the fact that the Jacobi symbol (q/N) is defined through the factorization of N, (q/N) is computable in polynomial time even when the factorization of N is not known !

It is easy to see, from the definition of the Jacobi symbol and the one of a quadratic residue, that if (q/N) =-1 then q must be a quadratic non residue mod N. In fact, q must be a quadratic non residue either mod p_1 or mod p_2. However, if $(q/N) = +1$, then either q is a quadratic residue mod N or q is a quadratic non residue for both the prime factors of N.

Let us count how many of the q's, such that $(q/N) = 1$, are quadratic residues.

Theorem: Let p be an odd prime. Then Z_p^* is a cyclic group.

Theorem: Let g be a generator for Z_p^*, then g^s mod p is a quadratic residue iff s is even.

Corollary: Half of the numbers in Z_p^* are quadratic residues and half are quadratic non residues.

Corollary: Let $N = p_1 * p_2$ where p_1 and p_2 are odd primes. Then half of the numbers in Z_N^* have Jacobi symbol equal to -1 and thus are quadratic non residues. The Jacobi symbol of the rest of the numbers is 1. Exactly half of these latter ones are quadratic residues.

2.2 A difficult problem in Number Theory.

If the factorization of N is not known and $(q/N) = 1$, then there is no known procedure for deciding whether q is a quadratic residue mod N (i.e. if the equation $q = x^2$ mod N is solvable). Such a decision problem is well known to be hard in Number Theory. A polynomial solution for it would imply a polynomial solution to other open problems in Number Theory, for example deciding whether a composite N, whose factorization is not known, is the product of 2 or 3 primes, see open problems 9 and 15 in Adleman [a].

2.3 A number theoretic result.

We want to show that deciding whether q is a quadratic residue mod N, is not hard in some special cases, but is **hard on the average** in a very strong sense.

Let us recall the weak law of large numbers:

If $y_1, y_2, \ldots, y_k$ are k independent Bernoulli variables such that $y_i = 1$ with probability p, and $S_k = y_1 + \ldots + y_k$, then for real numbers ψ, $\delta > 0$, $k \geq 1/4\delta\psi^2$ implies that $\Pr(|(S_k/k) - p| \geq \psi) \leq \delta$.

Notice that k is bounded by poly$(1/\psi, 1/\delta)$.

Set $A_N^* = \{ x \mid x \in Z_N^* \text{ and } (x/N) = 1 \}$.

Definition: For a composite number N, and for real $\varepsilon > 0$, we say that we can guess with ε advantage whether q drawn at random from A_N^* is a quadratic residue mod N if we can guess, in polynomial($|N|$) time, quadratic residuosity mod N correctly for at least $(50 + \varepsilon)\%$ of the $q \in A_N^*$.

Theorem 1: Let $q \in A_N^*$. For real numbers ε, $\delta > 0$, if we could guess, with an ε advantage whether q, drawn at random, is a quadratic residue mod N, then we could decide quadratic residuosity of any integer mod N with probability $1 - \delta$ by means of a polynomial($|N|$, $1/\varepsilon$, $1/\delta$) time probabilistic algorithm.

Proof: Let $\varepsilon = 1$. Assume to the contrary that we have a polynomial time magic box MB which guesses correctly whether $q \in A_N^*$ is a quadratic residue mod N, 51 times out of 100. Let α and β be the below defined conditional probabilities:

α = Pr(MB answers "q is a quadratic residue" /q is a quadratic residue mod n)

β = Pr(MB answers "q is a quadratic residue" /q is a quadratic non residue mod

N, $q \in A_N{}^*$).

Notice that, in order for MB to have a 1% advantage, it must be that $|\alpha - \beta| \geq 2/100$! Construct a sample of k quadratic residues chosen at random in $Z_N{}^*$, (the value of k will be defined later on). This can be easily done by picking $s_1,...,s_k$ at random in $Z_N{}^*$ and squaring them mod N.

Prepare two counters R and NR.

Feed each $s_i{}^2$ to MB. Every time that MB answers "quadratic residue", increment the R counter. Every time that MB answer "quadratic non residue", increment the NR counter.

If k is chosen to be suitably large (but still "reasonably small" !) the weak law of large numbers assures that $Pr(|\alpha - R/k| > 2/300) < 0.5*10^{-6}$; i.e. R/k is a very good approximation to how well MB guesses if the inputs are only quadratic residues. Note that α need not be equal to 51/100.

Let now q, be an element of $Z_N{}^*$ that we want to test for quadratic residuosity. Generate $x_1,...,x_k$ quadratic residues at random in $Z_N{}^*$ and compute $y_i = q*x_i$ for i =1,...,k. Notice that

a) if q was a quadratic residue, then the y_i's are random quadratic residues in $Z_N{}^*$

b) if q was a quadratic non residue in $A_N{}^*$, then the y_i's are random quadratic non residues in $A_N{}^*$.

Let us postpone the proof of (a) and (b) and assume, for the time being, that they are true. Feed MB with the sample $\{y_i\}$ and increment the counter R and NR initially set to 0.

If $|\alpha - R/k| < 2/300$, then with probability $1\text{-}10^{-6}$ q was a quadratic residue mod N, otherwise, again with probability 1 -10^{-6}, q was a quadratic non residue mod N.

What remains to be proved is (a) and (b). We will only prove (a). It will be enough to prove that, given **any** quadratic residue q, **any** other quadratic residue y in Z_N^* can be written as y =q*x where x is a quadratic residue mod N. It is a well known theorem in algebra that $Z_N^* = Z_{p_1}^* * Z_{p_2}^*$. Thus let a and b be, respectively, generators for $Z_{p_1}^*$ and $Z_{p_2}^*$. Then any element of Z_N^* can be written uniquely as $a^i b^j$ where $1 \le i \le p_1$-1 and $1 \le j \le p_2$-1. Moreover q is a quadratic residue mod N iff it can be written as q = $a^{2i} b^{2j}$ where again $1 \le 2i \le p_1$-1 and $1 \le 2j \le p_2$-1. Thus if y is any other quadratic residue, y =$a^{2s} b^{2t}$; then by setting x = $a^{2(s-i)} b^{2(t-j)}$ part (a) is proved.

Theorem 2: Let $q \in A_N^*$. Let r be a given quadratic non residue mod N, such that $r \in A_N^*$. For real numbers ε, $\delta > 0$, if we could guess with an ε advantage whether q, drawn at random, is a quadratic residue mod N, then we could decide quadratic residuosity of any integer mod N with probability 1- δ by means of a polynomial(|N|, 1/ε, 1/δ) time probabilistic Algorithm.

Proof: A little care is needed for theorem 2, which is, different from theorem 1. Here we know some extra information: namely that r is a quadratic non residue mod N whose Jacobi symbol is 1. We must show that this extra information cannot help us to decide quadratic residuosity mod N in polynomial time.

Let ε=1. Assume that given **any** r quadratic non residue mod N, $r \in A_N^*$, someone could build a polynomial time magic box MB_r that has a 1% advantage in distinguishing between quadratic residues and non residues mod N. Then we will show that even if one is not given such an r, he could still decide quadratic residuosity in the following way. Construct set T consisting of 20 elements chosen at random from A_N^*. With probability 1- $(1/2)^{20}$ one of the elements in T will be a quadratic non residue mod N. For each $x \in T$ do the following:

Choose k as in theorem 1. Construct MB_x and test its performance on k random quadratic residues, $S=\{s_1, \ldots, s_k\}$, as we did in Theorem 1. Also pick $y_1, \ldots, y_{20}$ at random from A_N^*. Again, with very high probability, at least one of the y_i's will be a quadratic non residue. Now, construct samples $H_i=\{y_i\, s \mid s \in S\}$, and feed them into MB_x.

If MB_x performs on all the H_i's exactly as it performed on S, then MB_x can not decide quadratic residuosity and x was a quadratic residue. Go to the nest element in T.

If MB_x performs clearly differently on, say H_i, than on S, then y_i is a quadratic non residue and, most importantly, we got a magic box, MB_x, which distinguishes between quadratic residues and non residues in polynomial time. This will clearly happen when we build MB_x, $x \in T$, where x is a quadratic non residue mod N. Thus we derive a contradiction with our assumption that deciding quadratic residuosity is hard.

In the above, we assumed that given **any** quadratic non residue r, $r \in A_N^*$, someone was able to construct a magic box MB_r, having a 1% advantage in deciding quadratic residuosity, and we derived a contradiction.

Suppose one is able to build a MB_r, having a 1% advantage in deciding quadratic residuosity, only for 1% of the quadratic non residues, r, $r \in A_N^*$. Then all that has to be changed in the above proof is to increase the size of the set T, so that T will include a suitable r.

3. DO WE ALREADY HAVE A WAY TO SEND ENGLISH MESSAGES IN A PUBLIC KEY CRYPTOSYSTEM ? *

In what follows n is a composite number product of two large odd primes, p_1 and p_2. The Rabin's function f, $f: Z_n \to Z_n$, is so defined: $f(x) = x^2$ mod n.

Notice that f is a 4-to-1 function because of our choice of n; in fact a quadratic residue q mod n has 4 square roots mod n (2 if we disregard minus signs) x, -x mod n, y, -y mod n. The following theorem shows how hard is to invert f.

Theorem (Rabin): If for 1% of the quadratic residues q mod n one could find one square root of q, then one could factor in Random Polynomial Time.

The theorem follows from the following lemma that we state without proof.

Lemma: Let q be a quadratic residue mod n. If we knew x and y, 2 square roots of q mod n such that $x \neq y,-y$ mod n, then we could easily factor n. (In fact the greatest common divisor of n and x+y is a factor of n).

Quick proof of Rabin's theorem: Assume that we have a magic box M such that given q, a quadratic residue mod n, for 1% of the q's it outputs one square root of q mod n. Then we could factor n by iterating the following step:

> Pick i at random in Z_n^* and compute $q = i^2$ mod n. Feed the magic box M with q. If M outputs a square root of q different from i or -i mod n, then (by the above lemma) factor n.

The expected number of iterations is low as, at each step, we have 0.5% chances to factor n.

The Rabin's function can be used to build the following public key cryptosystem. Any user in the system publicizes a composite number product of two

* The result in this section has been obtained in collaboration with Vijay Vazirani.

large primes. Let n be the number relative to user A. Define $E_A(x)$ to be x^2 mod n. As A knows the factorization of n, he could compute the 4 square roots of m^2 mod n and get the message m. The ambiguity in the decoding could be eliminated, for example, by sending the first 20 digits of m in addition to m^2 mod n; notice that this extra information cannot effectively help in factoring: we could always guess the first 20 digits of m.

The proof, so far accepted, that this public key cryptosystem is as hard to break as factoring, can be sketched in the following way: whoever can get a message m back from its encryption m^2 mod n 1% of the times, is actually realizing the magic box of the above theorem and thus could factor n.

We would like to point out the following fact.

Claim: If M, the set of messages, is "sparse" in Z_n, then the theorem of Rabin does not imply that decoding is as hard as factoring.

By "sparse" we mean that choosing at random $x \in Z_n$, the probability that x is a message is virtually 0. We will see that is the case for the ASCII code representation of English sentences.

Proof: Assume that we are able to invert the Rabin's function f only on f(M). Then we would have a magic box MB such that, fed with m^2 mod n, outputs m for all $m \in M$; and, fed with q, outputs nothing whenever q $\neq$ m^2 mod n for all $m \in M$ (except, at most, for a negligible portion of the q's). With the use of such an MB we could decode but not factor ! Let us follow the above proof for such an MB. If we pick $m \in M$ and feed MB with m^2 mod n, then we get back m and we cannot factor. If we pick i not belonging to M and we feed MB with i^2 mod n, the the probability that one square root of i^2 mod n, different from i, belongs to M is 0 and we get no answer.

Remark: ASCII English is sparse. Hint: the average size of a word in an English dictionary is, say, 10. There are 25^{10} 10-long strings of letters, but there are only 10^4 words in an English dictionary. Thus, so far, we do not have a scheme for sending English sentences in a **provably secure** way in a Public Key Cryptosystem.

Remark: The following "philosophical" objection can be raised against the above magic box MB.

" It is impossible that a machine, given q as an input, outputs m if q = m^2 mod n for some $m \in M$ and nothing (except for a negligible number of cases) otherwise. In fact messages have MEANING, a completely extraneous concept to a machine ".

Such a "philosophical" statement is of course complexity-independent, thus it could be rejected if we can exhibit an exponential time machine M which does the job. Let M be the ASCII code representation of English sentences. Find, by an exhaustive search, the square roots of m^2 mod n. If one of them is the ASCII representation of a string of words in an English dictionary (no meaning is involved: "runningboxhorse" would do) output it.

Nothing prevents the fact that M could have an equivalent M' which runs in polynomial time. In other words, ASCII English, being sparse, has certain redundancies which could unable to find one square root of x^2 mod n quickly if we knew that it must be an ASCII English one !

5. HOW TO SEND ENGLISH MESSAGES IN A PUBLIC KEY CRYPTOSYSTEM IN A PROVABLY SECURE WAY

We want to show how the results in the previous sections provide a solution for sending securely, in a Public Key Cryptosystem, messages belonging to a sparse set in Z_N. Let us consider the ASCII English case.

Send, bit by bit, an English sentence in ASCII by the method described in section 3.

Remark: The transmission can be done 8 times faster by using an ASCII table for words instead of letters.

We now address the question of the security of the newly proposed Public Key Cryptosystem. Let $E(x)$ stand for our new encryption function and let M be the set of all possible messages. First, note that even if an eavesdropper guesses what a message is, he can not verify it (e.g to verify that q, the encoded i-th bit of $m \in M$, represents a 0, one must exhibit $x \in A_N^*$ such that $x^2 = q \bmod N$). However, the possibility of *understanding* a message without being able to prove what it is, is also dangerous for the security of the public key Cryptosystem. We show that, given $E(m)$ for $m \in M$, if an eavesdropper can do better than guessing m at random, then deciding quadratic residuosity of any integer mod N, is easy.

Recall that $A_N^* = \{x \in Z_N^* \mid (x/N) = 1\}$.

Definition: Let $x \in A_N^*$. The **signature** of x, $\sigma_N(x)$ is defined as,

$$\sigma_N(x) \leftarrow \begin{cases} 1 \text{ if } x \text{ is quadratic residue } \bmod N \\ 0 \text{ if } x \text{ is quadratic non residue } \bmod N \end{cases}$$

Let S_N^n be the set of all n-long sequences of elements from A_N^*.

Definition: Let $s \in S_N^n$, $s = (x_1, \ldots, x_n)$. The **n-signature** of s, $\Sigma_N(s)$, is defined as, $\Sigma_N(s) = \sigma_N(x_1)\ \sigma_N(x_2)\ \cdots\ \sigma_N(x_n)$

Definition: A **decision** function is a function $d: S_N{}^n$ --> $\{0,1\}$.

Let $a=(a_1, \ldots, a_n)$, $b=(b_1, \ldots, b_n)$ be n-signatures.

Definition: We say that a and b are **adjacent** if and only if there exists a k, $1 \le k \le n$ such that $a_k \ne b_k$ and for all $i \ne k$ $a_i = b_i$. The **distance** between a and b is defined as: distance(a, b) = the number of positions i such that $a_i \ne b_i$.

For any decision d and n-signature l, let $P_d(l)$:$\{0,1\}^n$--> $[0,1]$ be defined as

$P_d(l)$ = probability ($d(x)=1 \mid \Sigma_N(x) = l$ for $x \in S_N{}^n$).

Theorem 3: If there exists a decision function d which is easy to compute and two n-signatures u and v, such that $|P_d(u) - P_d(v)| > 1/100$, then deciding quadratic residuosity is easy.

Proof: Suppose there exists a decision d and two n-signatures u and v such that $|P_d(u) - P_d(v)| > 1/100$. Let distance($u$, v) = m, and for $0 \le i < m$, let a_i's be n-signatures such that $a_0 = u$, $a_m = v$ and a_i is adjacent to a_{i+1} for all i. As $|P_d(u) - P_d(v)| > 1/100$, there must exist i, $0 \le i \le m-1$, such that $|P_d(a_i) - P_d(a_{i+1})| \ge 1/100n$. For convenience let $s = a_i$ and $t = a_{i+1}$.

Let us choose $\psi = 1/3(1/100n)$ and $\varepsilon = 0.5\ 10^{-6}$. By the weak law of large numbers compute a sample size k, $k \le$ polynomial($1/\psi$, $1/\varepsilon$), such that if we choose k elements, $x_1, \ldots, x_k$ at random from $A_s = \{x \in S_N{}^n \mid \Sigma_N(x) = s\}$ and k elements, $y_1, \ldots, y_k$ at random from $A_t = \{x \in S_N{}^n \mid \Sigma_N(x) = t\}$, then

Pr ($P_d(s) - (d(x_1) + \ldots + d(x_k))/k > 1/\psi) < \varepsilon$ and

Pr ($P_d(t) - (d(y_1) + \ldots + d(y_k))/k > 1/\psi) < \varepsilon$

As $s=(s_1, \ldots, s_n)$ and $t=(t_1, \ldots, s_n)$ are adjacent, let us assume, without loss of generality, that for all $i = 1, \ldots, r-1, r+1, \ldots, n$, $s_i = t_i$ and $s_r = 1$, $t_r = 0$.

We will now show that we can decide quadratic residuosity mod N with probability greater than $1-1/10^6$. Let q be an element of A_N^* that we want to test for residuosity. Choose k random quadratic residues in A_N^*: $x_1^2, \ldots, x_k^2$ and compute $Y_j = q\ x_j^2 \bmod N$ for $1 \le j \le k$. By theorem 1, the Y_j's are all quadratic residues if q is a quadratic residue, and all quadratic non residues in A_N^*, otherwise.

In theorem 2 we showed that the knowledge of a non residue in A_N^* does not help in deciding quadratic residuosity. Therefore we can assume that such a non residue, h, is known, which allows us to pick quadratic non residues at random from A_N^* (compute $h \cdot x^2$).

We are now ready to decide whether q is a quadratic residue.

(* construct a random sample, SAMPLE, of k elements in S^n such that SAMPLE= $\{ (y_{j,1}, \ldots, y_{j,n}) \in S_N^n \mid$ for all $1 \le i \le n,\ i \ne r,\ 1 \le j \le k\ \sigma_N(y_{j,i})=s_i$ and $y_{j,r}=Y_j \}$ of

*)

For $i = 1, \ldots, r-1, r+1, \ldots n$ do

begin

For $j = 1, \ldots, k$ do

draw $x \in A_N^*$ at random.

if $s_i = 1$ then $y_{j,i} = x^2 \bmod N$

else if $s_i = 0$ then $y_{j,i} = h\ x^2 \bmod N$

end.

(* Evaluate the decision function d on each each member of the sample *)

For $j = 1, \ldots, k$ do

begin

$X_j = d(y_{j,1}, \ldots, y_{j,r-1}, Y_j, y_{j,r+1}, \ldots, y_{j,n})$

end.

Notice that the entire sample $\{y_{j,1}, \ldots, y_{j,r-1}, Y_j, y_{j,r+1}, \ldots, y_{j,n} \mid 1 \le j \le k\}$ is either a subset of A_s or a subset of A_t. Thus with probability $1-\varepsilon$ one of these two mutually exclusive events will occur

(1) $|(X_1+\ldots+X_k)/k - P_d(s)| < 1/300n$

or,

(2) $|(X_1+\ldots+X_k)/k - P_d(t)| < 1/300n$

If case (1) occurs we conclude with probability greater than $1-2\varepsilon = 1-10^{-6}$ that q is a quadratic residue, else we conclude, again with probability greater than $1-10^{-6}$ that q is a quadratic non-residue.

Let us extend the notion of a discriminating function so that the function can take on more than 2 values. For any non empty set A, let $D:S_N{}^n \rightarrow A$. Let $a \in A$, then $P_{D,a}(l)=$ probability$(D(x)=a \mid \Sigma_N(x)=l$ for $x \in S_N{}^n)$ The following theorem is an easy extension of theorem 3 and we will state it without proof.

Theorem 4: If there exists a discriminating function $D:S_N{}^n \rightarrow A$, $a \in A$ and 2 n-signatures u and v such that $|P_{D,a}(u)-P_{D,a}(v)|>1/\varepsilon$, then deciding quadratic residuosity mod N is easy.

The next theorem takes us back to messages. But first, some more notation must be introduced. Let $M^n = \{ m_1, m_2, \ldots\}$ be the set of messages whose length is n, $n \le p(|N|)$ where p is a polynomial. Set $k=|M^n|$. Let M_i be the set of all possible encodings of message i. Clearly, $M_i \subseteq S_N{}^n$ and for all i and j, $|M_i|=|M_j|$, and thus $|M^n|= k\,|M_i| =|M|$. Let MB be a magic box that receives as input $E(m)$ for $m \in M^n$, and guesses $1 \le i \le t$ such that $m_i=m$. Let $r_{i,j}$ denote the number of encodings of message m_j, on which MB answers i. Clearly, $r_{i,i}$ will denote the number of times, over all possible encodings of m_i, that MB answers correctly.

Theorem 5: Let $\varepsilon < 1-1/k$ be a non negligible positive number. If $\sum_i r_{i,i}/kM > \varepsilon + 1/k$, then deciding quadratic residuosity mod N is easy.

Proof: By assumption $\sum_i r_{i,i} > \varepsilon kM + M$.

Claim: There exist two messages m_i, m_j such that $r_{i,i} - r_{i,j} > \varepsilon M$.

Proof: Assume, to the contrary, that for all $i \neq j$, $r_{i,i} - r_{i,j} \leq \varepsilon M$. Then $kM = \sum_j \sum_i r_{i,j} \geq \sum_i (r_{i,i} + (k-1)r_{i,i} - (k-1)\varepsilon M) = \sum_i (kr_{i,i} - (k-1)\varepsilon M) >$ (by hypothesis) $-k(k-1)\varepsilon M + k^2 \varepsilon M + kM = kM + k\varepsilon M > kM$. Contradiction.

Let us transform MB into a discriminating function $D: S_N{}^n \rightarrow M^n \cup \{\delta\}$. If $x \in S_N{}^n$, and MB, on input x, outputs j, then set $D(x) = m_j$. If y is not an encoding of any message, then one of 3 cases must occur:

1. MB outputs $1 \leq i \leq t$. Set $D(y) = m_i$.
2. MB outputs $i < 1$ or $i > t$. Set $D(y) = \delta$.
3. MB does not answer within a certain time limit. Set $D(y) = \delta$.

Now, note that in the claim just proved, we showed that for such a decision function, there exist m_i, m_j such that $|P_{D,m_i}(m_i) - P_{D,m_i}(m_j)| > \varepsilon$. Thus the hypothesis of theorem 4 holds, and deciding quadratic residuosity mod N is easy.

Theorem 5 shows that inverting the function E on the encrypted messages is as hard as deciding quadratic residuosity, independently of the sparseness of M^n.

Bibliography

[1] Adleman, L., *Private Communication*, 1981.

[2] Adleman, L., *On Distinguishing Prime Numbers from Composite Numbers*, Proceedings of the 21st IEEE Symposium on the Foundations of Computer Science (FOCS), Syracuse, N.Y., 1980.

[3] Blum, M., and Micali, S., *How to Flip A Coin Through the Telephone* , Extended Abstract, 1981.

[4] Diffie, W., and Hellman, M., *New Direction in Cryptography*, IEEE Trans. on Inform. IT-22, 6 (1976), 644-654.

[5] Miller, G., *Riemann's Hypothesis and Tests for Primality*, Ph.D. Thesis, U.C. Berkeley, 1975.

[6] Rabin, M., *Digitalized Signatures and Public-Key Functions As Intractable As Factorization*, MIT/LCS/TR-212, Technical Memo MIT, 1979.

[7] Rivest, R., Shamir, A., Ademan, L., *A Method for Obtaining Digital Signatures and Public Key Cryptosystems*, Communications of the ACM, Febuary 1978.

ACKNOWLEDGEMENTS

We would like to express our deep gratitude to Manuel Blum, our adviser, for having introduced us to this subject and for having been a continuous source of great ideas in so many insightful discussions.

A special thank is due to Vijay Vazirani for his help in the proof of section 3.

We also sincerely thank Richard Karp, Donald Johnson, David Lichtenstein, Jeff Shallit, Po Tong and Andy Yao for their generous help throughout this research.