A RANDOMIZED PROTOCOL FOR SIGNING CONTRACTS

(Extended Abstract)

S. Even*, O. Goldreich and A. Lempel

Computer Science Department
Technion - Israel Institute of Technology
Haifa, Israel

1. INTRODUCTION

Suppose two parties A, and B, in a communication network, have negotiated a contract, which they wish to sign. To this end, they need a protocol which has the two following properties:

(1) At the end of an honest execution of the protocol, each party has a signature of the other.

(2) If one party, X, executes the protocol honestly, his counterpoint, Y, cannot obtain X's signature to the contract without yielding his own signature.

It was shown by Even and Yacobi [1] that no such deterministic protocol exists without the participation of a third party. Assuming reliable third parties exist, it is still desirable to have a protocol for signing contracts in which no third party is required. Even [2], proposed a protocol based on the puzzle concept of Merkle [3] using any Public Key Cryptosystem (PKCS) deemed secure. Other protocols, relying on the infeasibility of certain number-theoretic operations, such as factoring of large integers, were suggested by Blum and Rabin [4] and Blum [5].

The notion of Oblivious Transfer (OT) was introduced by Rabin [6], with an implementation based on the integer factoring problem. We propose what we believe to be a more natural definition and present an implementation using any PKCS.

*Supported in part by the Fund for the Promotion of Research at the Technion

We describe a protocol for signing contracts which uses OT. Its advantage over the protocol proposed by Even [2] is that there is neither reliance nor reference to the value of the contract's context.

2. ASSUMPTIONS

We assume the existence of a secure PKCS [7] and that the cost and time of computation is approximately the same for both parties to the contract.

Let E_x and D_x be the encryption and decryption algorithms, respectively, generated by feeding the word x to the key generating algorithm. We assume that for every x and every ω, E_x and D_x are defined and

$$E_x(D_x(\omega)) = D_x(E_x(\omega)) = \omega.$$

We also assume that every participant A, in the network, randomly chooses a word x_A, from which he generates an encryption-decryption pair (E_{x_A}, D_{x_A}) (hereafter denoted by (E_A, D_A)) and announces his encryption key (E_A). Clearly, A can sign a document M by transmitting $D_A(M)$.

We also use a secure conventional cryptosystem F. Its existence is guaranteed by the (assumed) existence of a secure PKCS; however, one can use any trusted conventional system, e.g. the DES [8]. Denote the encryption and decryption algorithms with key k, by F_k and F_k^{-1} respectively.

3. OBLIVIOUS TRANSFER

An Oblivious Transfer (OT) of a recognizable message M is a protocol by which the sender (hereafter denoted by S) transfers to the receiver (hereafter denoted by R) the message M, so that R can read M with probability one half while S has no way of knowing whether R can actually read M.

Formally OT has to satisfy the following axioms:

(i) R can recognize M [e.g. M is a signature on some known message M', i.e. $M = D_S(M')$].

(ii) If S is honest R gets M with a priori probability one half. For S, the posteriori probability that M was actually read by R remains one half.

(iii) If S tries to cheat, R will detect it with probability at least one half.

An implementation of an OT satisfying these axioms is presented in Section 6.

4. THE CONTRACT SIGNING PROTOCOL

The parties to the protocol will be called A and B.

(1) A generates randomly an ordered set $\{x_i\}_{i=1}^n$ of keys for the conventional system F. He declares that if B is able to present (n-m) signed members of the ordered set $\{M_i\}_{i=1}^n$, then he is committed to the contract C, and signs this declaration. B acts symmetrically generating the keys $\{y_i\}_{i=1}^n$.

(2) A transmits to B the ordered set $\{F_{x_i}(D_A(M_i))\}_{i=1}^n$.
B transmits to A the ordered set $\{F_{y_i}(D_B(M_i))\}_{i=1}^n$.

(3) for i = 1 to n do
begin
A sends x_i to B via OT
B sends y_i to A via OT
end

(4) for j = 1 to ℓ do (ℓ is the length of the keys for F)
begin
A transmits the j-th bit of every x_i to B
B transmits the j-th bit of every y_i to A
end

Note: The interleaving in step (3) is not essential.

To avoid being cheated the parties should take the following precaustions:

(a) During step (3) each party, while playing the role of R in OT, should use the cheat-detection mechanism of the OT. [Its existence is guaranteed by axiom (iii). Also note that the keys are recognizable using the information transferred in step (2).]

(b) While executing step (4) each party should check whether the bits revealed to him during the alternating substeps match the bits of the keys actually disclosed to him in step (3). [Note that after step (3) is completed, each party knows, on the average, one half of his counterpart's keys; the latter, however, is oblivious as to which of his keys were actually disclosed.]

A party will stop further execution of the protocol as soon as he detects an attempt to cheat.

5. ANALYSIS OF THE PROTOCOL

If both parties follow the protocol honestly to its conclusion then either will have a signature by the other to the contract C, and will know it. In fact, each party will have all n signed M_i's.

Let PR(n,m) denote the probability that X gets at least n-m keys during the execution of step (3) of the protocol.

Theorem: If $n \geq 100$ and $n-m \geq .78 \cdot n$ then $PR(n,m) < 2^{-(m+1)}$.

Thus, the probability that X will have his counterpart's signature to the contract before the execution of step (4) of the protocol, is less than $(\frac{1}{2})^{m+1}$. (If this occurs X might stop the procedure before his counterpart has X's signature.)

If X decides to cheat Y, he has to make sure that Y gets less than (n-m) signed M_i's. To this end, during the execution of the protocol, X must designate at least m+1 M_i's for which Y is not to have X's signature.

Without loss of generality, assume that X = A. A may prevent B from having the i-th signature (i.e. $D_A(M_i)$) by one of the following actions:

(1) Transfer a "fake" $F_{x_i}(D_A(M_i))$ in step (2).

(2) Cheat in execution of the OT of x_i (in step (3)).

(3) Cheat in the disclosure of the bits of x_i (in step (4)).

Clearly, it makes no sense to take more than one of these three possible actions, since one of the first two suffices to make sure tha B will not get $D_A(M_i)$, while a multiple attempt for the same i may increase the chances of being caught.

By axioms (i) and (iii) of OT, actions (1) or (2) will be detected with probability at least one half; while by axiom (ii), the probability of being caught in action (3) is exactly one half. Thus, the probability that any party will succeed in cheating the other, is at most $(\frac{1}{2})^{m+1}$.

The total risk for X in using the protocol amounts to the sum of two probabilities; the probability that Y gets the signature in step (3) and the probability that Y succeeds in cheating X. This risk is bounded from above by $2(\frac{1}{2})^{m+1} = (\frac{1}{2})^{m}$.

An important feature of our protocol is that with high probability, $(1-2^{-m})$, the feasibility of obtaining a signature by computation

is about the same for both parties. This observation is based on the fact that computing the signature becomes feasible only during the execution of step (4) and at this point each party knows that, with very high probability, he has the information required for the computation of his counterpart's signature. This feature is absent from Even's protocol [2], where the information required for the computation of the signature passes from X to Y, before X is able to verify that he can compute the signature of Y.

It should be noted, however, that if X stopes the correspondence during step (4), his advantage over Y is at most one bit per key. If this is considered too big an advantage, one can change step (4) of the protocol so that only a single bit is transferred at a time instead of n bits.

6. AN IMPLEMENTATION OF OBLIVIOUS TRANSFER

The proposed implementation of an oblivious transfer of a message M from S to R proceeds as follows:

(0) S chooses, randomly, two pairs $\{(E_i,D_i)\}_{i=1}^{2}$ of encryption-decryption algorithms for the PKCS. R chooses, randomly, a key K for the conventional cryptosystem F.

(1) S transmits E_1, E_2 to R.

(2) R chooses, randomly, $i \in \{1,2\}$ and transmits $E_i(K)$ to S.

(3) S chooses, randomly, $j \in \{1,2\}$, computes $K' \triangleq D_j(E_i(K))$ and transmits the pair $(F_K,(M),j)$ to R.

Remarks:

(1) Assuming that K looks like random noise and that E_1, E_2 have the same range, S cannot know (or guess with probability of success greater than one half) whether K', computed by him, is the K choosen by R.

(2) By the assumption that the PKCS is secure, R cannot find K' when $K' \neq K$. Due to the security of the conventional cryptosystem, R must know K' in order to read M.

(3) R can read M iff $i = j$. Thus, he can detect cheating by S with probability one half.

(4) In the RSA [9] scheme, distinct E_i's do not have the same range; nevertheless, this can be fixed. Other implementations of OT via RSA were suggested by Rabin and Micali.

7. COMMENT ON EXTENSIONS

Using an (n-m,n) threshold - scheme and generalizing the use of the OT we have developed protocols for:

(1) Sending Certified Mail [Mailing Disclosures],

(2) Coin Flipping [Lottery],

with the same exponentially decreasing probability of being cheated.

REFERENCES

[1] Even, S., and Yacobi, Y., Relations Among Public Key Signature Systems, TR#175, Computer Science Dept., Technion, Haifa, Israel, March 1980.

[2] Even, S., A Protocol for Signing Contracts, TR#231, Computer Science Dept., Technion, Haifa, Israel, January 1982.

[3] Merkle, R.C., Secure Communication Over Insecure Channel, Comm. ACM, Vol. 21, April 1978, pp. 294-299.

[4] Blum, M., and Rabin, M.O., How to Send Certified Electronic Mail. In preparation.

[5] Blum, M., How to Exchange (secret) Keys, Memo No. UCB/ERL M81/90, March 1982. To appear in CACM.

[6] Rabin, M.O., Private communication.

[7] Diffie, W., and Hellman, M.E., New Directions in Cryptography, IEEE Trans. Inform. Theory, Vol. IT-22, No.6, November 1976, pp. 644-654.

[8] Data Encryption Standard, National Bureau of Standards, Federal Information Processing Standards, Publ. 46, 1977.

[9] Rivest, R., Shamir, A., and Adleman, L., A Method for Obtaining Digital Signatures and Public Key Cryptosystems. Comm. ACM, Vol 21, February 1978, pp. 120-126.