Unconditionally-Secure Digital Signatures

David Chaum
Centre for Mathematics and Computer Science (CWI)
Kruislaan 413, 1098 SJ Amsterdam

Sandra Roijakkers
Eindhoven University of Technology (TUE)
Den Dolech 2, 5600 MB Eindhoven (Research conducted at CWI)

Abstract. All known digital signature schemes can be forged by anyone having enough computing power. For a finite set of participants, we can overcome this weakness.

We present a polynomial time protocol in which a participant can convince (with an exponentially small error probability) any other participant that his signature is valid. Moreover, such a convinced participant can convince any other participant of the signature's validity, without interaction with the original signer.

An extension allows, in most cases, a participant who receives a signature from any source to convince each other participant of its validity. If a participant cannot use the signature to convince others, he knows so when he receives it.

1. Introduction

For many purposes digital signatures, as originally proposed in [DH76], are a useful tool: everyone can test the validity of a signature and no one can forge one. Some well-known examples are the RSA-scheme ([RSA78]), the scheme of ElGamal ([ElG85]), and the Fiat-Shamir scheme ([FS87]). They all rely on cryptographic assumptions, which means forgery is, in principle, always possible by someone using enough computing power.

The scheme we present here does not have this disadvantage: even someone with infinite computing power is unable to forge a signature. The price we pay for this is that our scheme has to be set up among a fixed, finite set of participants.

In [RB89], a "non-cryptographic weaker version of digital signatures" was introduced. Their model is essentially the same as ours. Unlike a digital signature, however, their scheme requires each recipient to conduct a protocol with all other participants to test a signature after it has been issued to him. Even so, only a participant who receives a signature directly from the signer can convince another

A.J. Menezes and S.A. Vanstone (Eds.): Advances in Cryptology - CRYPTO '90, LNCS 537, pp. 206-214, 1991.

participant. In contrast, our result is a true signature scheme: a signature is a collection of bits which can be verified independently by any recipient and successively transferred to other participants who can do the same (a related work based on cryptographic assumptions is described in [BPW90]).

In establishing the public key in our scheme, the signer receives messages from the other participants by an untraceable sending protocol, like that introduced as "the Dining Cryptographers Problem" ([Cha88]). The signature will contain some of the values that were received. The essential observation is that since the signer does not know who sent what, he will be unable (except with very small probability) to give a signature that one participant will accept that will not similarly be accepted by any participant.

Section 2 describes the model and the assumptions we make, and gives a precise definition of our aim. In Section 3, a known untraceable sending scheme is introduced. Here we also explain how to make it impossible for a disruptive participant to change (one of) the numbers sent by someone else without being caught. Section 4 explains the basic protocol, and shows that it achieves the desired result. In Section 5, we extend the protocol such that a first receiver can convince a second receiver, and that each participant who receives the signature later on knows *a priori* if he can convince a next receiver. In the last section some final remarks will be made.

2. Setting and Objectives

In the first subsection we give the setting we work in; in the second subsection the objectives are given.

2.1. Model and Assumptions

We assume that the "world" consists of a finite set $\mathcal{P}$ of n participants $\{P_1, P_2, \ldots, P_n\}$.

We assume the following means of communication between participants:

1. *an authenticated broadcast channel.* This enables each participant to send the same message to all other participants, identifies the sender, and is completely reliable. In particular, if any participant receives a message via the broadcast channel, all other participants will receive the same message at the same time.

2. *a private, authenticated channel between each pair of participants*. Such a channel cannot be read or tampered with by other participants, and each of the communicants is absolutely sure of the identity of the other.

2.2. Objectives

We want a participant S to be able to send a bit b to a participant R such that the following conditions hold:

1. only R receives b.
2. R can prevent S from convincing other participants that he sent $b\oplus 1$.
3. R can convince any participant that he got b from S.
4. R cannot convince any participant that he got $b\oplus 1$ from S.

Our aim is to obtain the four conditions, using a protocol that is polynomial time in a security parameter m, but with an error probability that is exponentially small in m, and we do not require any limitations on the computing power available to each participant.

Notice that in models which rely on cryptographic assumptions there is a difference between cheating that can be done offline at home, without risk of detection, and cheating which involves a substantial probability of being detected. Since we have no such assumptions, in our model there is no difference between the two.

3. Background

We make use of an Untraceable Sending protocol as introduced by [Cha88] and further elaborated by [BB89] and [Pfi89].

This protocol allows participants to send messages (elements of the abelian group $(\mathbb{Z}_v, +)$) to a fixed participant S, such that S does not know who sent which message.

The protocol relies on the use of keys. A key is a random group element, known by two participants. One of them uses the key, the other the negative (modulo v) of the key ([BB89], [Pfi89]).

Every participant, except S, broadcasts the sum (mod v) of his message and all keys he shares. Only one participant, who is allowed to send, has a message unequal to 0. Now S computes the sum (mod v) of all sums broadcasted and his own sum. This equals the message, because all keys add up to 0.

It is not necessary that each pair of participants shares a key; even if a particular participant shares only one key that another participant does not know, that participant cannot compute the first participant's message. Let the participants share keys according to a trust graph T on the participants: if (P_i, P_j) is an edge in T, P_i and P_j trust each other, and they share a key.

We start with a completely connected graph T. If a particular round is disrupted (how to detect this will follow from Section 4), this round is entirely opened (i.e. all secret messages and keys are broadcasted), resulting in disagreement about a key or in detection of a participant who disrupted. In the first case, the corresponding edge in T is removed; in the second the corresponding vertex of T is removed.

Thus, opening a disrupted round always results in a reduction of T, and only vertices corresponding to disrupters and edges connected to at least one disrupter are removed.

4. Basic Protocol

We want some participant $S \in \mathcal{P}$ to send a random bit b, with his signature attached to it, to some participant $R \in \mathcal{P}$. To achieve this, all participants initially agree on a security parameter m, such that $\frac{1}{2}\, m \cdot (0.65)^m$, upper bounding the error probability, is sufficiently small, and they agree on a prime p, $2^m < p < 2^{m+1}$.

First we need a *preparation phase* in which each of the $n-1$ participants unequal to S sends untraceably m pairs of random numbers to S.

Round μ of the preparation phase ($1 \leq \mu \leq m$) looks like:

step 1. The participants start with a subprotocol, called the *reservation phase*, to determine the order in which they have to send their messages. This subprotocol is described in [Cha88].

If it is not successful, this can be due to collision or to disruption. In the first case the participants just start again with the reservation phase; in the second, T is reduced before doing so.

After a successful reservation, the only thing each participant unequal to S knows is when he is allowed to send.

step 2. Each participant ($\neq$ S) sends S untraceably and in the defined order a pair of numbers (N^0, N^1) chosen uniformly from $\mathbb{Z}_p \times \mathbb{Z}_p$, and their product $C := N^0 \cdot N^1 \bmod p$. A disrupter can modify the pair by adding (modulo

p) some non-zero pair to it, but since S only accepts pairs $(\widetilde{N}^0, \widetilde{N}^1)$ for which the received $\widetilde{C}$ equals the product modulo p of $\widetilde{N}^0$ and $\widetilde{N}^1$, the probability that S accepts a modified pair is smaller than 2^{-m} (see below). If S does not accept a pair, this round is opened, T is reduced, and the participants start again with step 1.

After the preparation phase we can start with the *signing phase*:

S has obtained the $(n-1) \times m$ matrix A: $A_{ij} = (\widetilde{N}^0{}_{ij}, \widetilde{N}^1{}_{ij}, \widetilde{C}_{ij})$ for $1 \le i \le n-1$, $1 \le j \le m$. S only knows that each participant has sent him one entry of each column, while the participants distinct from S also know which entries of each column are theirs.

S sends his bit b to R by sending him b and the $(n-1) \times m$ matrix A^b: $A^b{}_{ij} = \widetilde{N}^b{}_{ij}$. R accepts this bit b if all the N^b he sent to S are correctly contained in this matrix.

R can convince an other participant P that he got b from S by sending him A^b. P accepts b from R (i.e. he is convinced that R accepted b from S) if he sees at least half of his N^b correctly in this matrix. If the protocol would require P to see all his N^b correctly, it would be rather easy for a disruptive S to convince R, while R could not convince anyone else.

One can compute the following error probabilities:

- $\mathbb{P}$(R has reason to reject a bit b that a non-disrupt S sent him)
  $= \mathbb{P}$(at least one N^b of R is disrupted, but still accepted by S)
  $$\le m \cdot \frac{1}{p-1}$$
  $\le m \cdot 2^{-m}$.

- Since in round μ of the untraceable sending protocol S receives $n-1$ $\widetilde{N}^b$, and sends a subset of them to R (after he received the whole set of $m \cdot (n-1)$ $\widetilde{N}^b$), we find:

  $\mathbb{P}$(R has reason to accept a bit b, which a non-disrupt P does not accept from him)
  $$= \sum_{k=0}^{\lfloor \frac{1}{2}m \rfloor} \binom{m}{k} \left(\mathbb{P}(\text{from a column of } A\text{, S sends both R's and P's } \widetilde{N}^b \text{ to R}) \right)^k$$

$$\Big(\mathbb{P}(\text{from a column of } A, \text{ S sends R's } \tilde{N}^{b} \text{ to R, but not P's } \tilde{N}^{b})\Big)^{m\text{-}k}$$

$$< \sum_{k=1}^{\lfloor \frac{1}{2}m \rfloor} \left(\frac{m+k}{k}\right)^{k} \left(\frac{m+k}{2m}\right)^{m} \left(\frac{m}{2m}\right)^{m}$$

$$< \tfrac{1}{2}\, m \left(\frac{3\sqrt{3}}{8}\right)^{m} .$$

- $\mathbb{P}$(P has reason to accept $b\oplus 1$ from (a disrupt) R, while R got b from S)
$$= \sum_{k=0}^{\lfloor \frac{1}{2}m \rfloor} \binom{m}{k} \left(\frac{p-1}{p}\right)^{k} \left(\frac{1}{p}\right)^{m-k}$$
$$< \left(\frac{1}{p}\right)^{\lceil \frac{1}{2}m \rceil} \cdot \tfrac{1}{2} \cdot 2^{m}$$
$$< 2^{\text{-}m}.$$

- $\mathbb{P}$(P has reason to reject b, while both S and R were honest)
$$< \binom{m}{\lceil \frac{1}{2}m \rceil} \left(\frac{1}{p}\right)^{\lceil \frac{1}{2}m \rceil}$$
$$< 2^{\text{-}m}.$$

As a consequence, an upper bound on the error probability is $\frac{1}{2}\, m{\cdot}(0.65)^{m}$.

5. Transferability

In this section we will denote the first receiver R by R_1 and the second receiver P by R_2.

At first glance, R_2 can pass the signature to a third receiver R_3, just as R_1 passed it to him. But the problem is that R_2 does not know how many of R_3's numbers in the matrix have been changed.

If only half of R_2's numbers in the matrix were correct, of course an honest R_2 does not want to guarantee that the same fraction (or more) of R_3's numbers are correct. Therefore, each time the signature is passed, the number of correct entries required has to be reduced.

This scheme has two disadvantages; m has to be very large if there are a lot of (potential) receivers, and each receiver has to know his position in the chain.

Therefore we adapt the scheme such that a receiver cannot change any of the $m \cdot (n-1)$ $\tilde{N}^b$ without being caught by the receiver he passes the signature to.

In the *preparation phase*, instead of sending (N^0, N^1, C), where $C = N^0 \cdot N^1 \bmod p$, each participant sends a set of four numbers (N^0, N^1, K, C), where N^0, N^1 and K are chosen uniformly from $\mathbb{Z}_p$, and C is their product $N^0 \cdot N^1 \cdot K \bmod p$. S only accepts triples $(\tilde{N}^0, \tilde{N}^1, \tilde{K})$ for which the received $\tilde{C}$ equals their product modulo p, thus the probability that he accepts a modified triple can easily be calculated to be smaller than 2^{-m}.

K is a key that determines a hash function H_K from a universal class of hash functions that map arbitrary large inputs to numbers of $\mathbb{Z}_p$. Given K, each participant knows H_K.

When S has obtained the $(n-1) \times m$ matrix A: $A_{ij} = (\tilde{N}^0{}_{ij}, \tilde{N}^1{}_{ij}, \tilde{K}_{ij}, \tilde{C}_{ij})$, the *signing phase* can start. To send a bit b to R_1, S sends him b and the $(n-1) \times m$ matrix A^b: $A^b{}_{ij} = (\tilde{N}^b{}_{ij}, X_{ij})$, where the checksums X_{ij} are defined by the following funtion on ordered sets:

$$X_{ij} := H_{\tilde{K}_{ij}}(\{\tilde{N}^b{}_{kl} \mid 1 \le k < n,\ 1 \le l \le m\} \cup \{X_{kl} \mid 1 \le k < n,\ 1 \le l < j\}) + \tilde{N}^{b\oplus 1}{}_{ij} \bmod p.$$

In words: the X_{ij} are output values of a hash function on all message numbers and all checksums of the previous columns. The random number $\tilde{N}^{b\oplus 1}{}_{ij}$ is added to make it impossible for a disruptive receiver who was not the sender of A_{ij} to calculate the key $\tilde{K}_{ij}$. Thus a receiver who modifies one of the $\tilde{N}^b{}_{ij}$ is unable to change the checksums accordingly. Therefore, a receiver who finds at least one of his checksums correct, knows that all message numbers are as S sent them to the first receiver.

R_1 checks that all the N^b he sent to S are correctly contained in A^b, and that all his checksums are correct. If something is wrong, R_1 knows S is a disrupter and he rejects the signature.

Each of the following receivers R_k ($k \geq 2$) checks if at least half of the N^b he sent to S are correctly contained in A^b. If this is not the case, R_k rejects the signature and knows that R_{k-1} or S has been cheating. Otherwise R_k checks his checksums. If at least one of them is correct, he assumes that all random numbers are as S gave them to R_1, as it is very unlikely that a preceding receiver modified them and guessed correctly a corresponding checksum. In this case R_k *strongly accepts* the signature, which means that he is convinced that receiver R_{k+1} will accept it from him. On the other hand, if none of R_k's checksums is correct, R_k has no idea of how many of an other participants' message numbers have been changed. Therefore he only *weakly accepts* the signature: he is convinced that S gave b to R_1, but he is not sure that he can convince someone else.

A receiver R_k can get additional information from the structure of the received matrix. For example, if R_k does not find all his N^b correctly in A^b, but at least one correct checksum on the values in the matrix, he knows that S has been cheating. The same holds if R_k finds an incorrect checksum that is the input of a correct checksum.

If a receiver would tell the next receiver which prefix of columns he can check that actually originate from S, it is even more often possible to point out a disrupter who is responsible for inconsistencies.

6. Summary and Suggestions for Further Research

We devised a signature scheme that is of polynomial complexity in the number of participants n and the security parameter m, allowing each of the participants to convince each other participant of the validity of his signature. Moreover, this convinced participant can convince every other participant of the signature's validity, without interaction of the original signer.

An extension of this scheme, that is still of polynomial complexity, allows a participant that receives the signature from any source to check *a priori* if he will be able to convince every other participant of the signature's validity (without interaction with the original signer).

For some applications (e.g. multiparty computations), it would be useful if non-acceptance of a signature from the original signer S, would enable the other participants to decide who is disrupting.

It would also be nice if a receiver of the signature, upon noticing inconsistencies, would know who has been cheating.

Finally, the efficiency of the scheme would be significantly improved if it was possible to sign arbitrary messages instead of single bits. Research is currently being done at CWI to use a universal hashing function on a message with the two numbers N^0 and N^1 as keys, instead of just choosing N^b for a single bit message b.

Bibliography

[BB89] J. Bos, B. den Boer: "Detection of Disrupters in the DC Protocol", *Advances in Cryptology - EUROCRYPT '89 Proceedings*, 1989.

[BPW90] G. Bleumer, B. Pfitzmann, M. Waidner: "A remark on a signature scheme where forgery can be proved", to appear in *Advances in Cryptology - EUROCRYPT '90 Proceedings*, 1990.

[Cha88] D. Chaum: "The Dining Cryptographers Problem: Unconditional Sender and Recipient Untraceability", *Journal of Cryptology 1/1*, 1988.

[DH76] W. Diffie, M.E. Hellman: "New Directions in Cryptography", *IEEE Transactions on Information Theory IT-22*, 1976.

[ElG85] T. ElGamal, "A Public Key Cryptosystem and a Signature Scheme Based on Discrete Logarithms", *IEEE Transactions on Information Theory IT-31*, 1985.

[FS87] A. Fiat, A. Shamir: "How To Prove Yourself: Practical Solutions to Identification and Signature Problems", *Proceedings of Crypto 86*, Lecture Notes in Computer Science, Vol. 263, Berlin: Springer - Verlag, 1987.

[Pfi89] A. Pfitzmann: "*Diensteintegrierende Kommunikationsnetze mit Teilnemher-überprüfbarem Datenschutz*", Dissertation Universität Karlsruhe, Fakultät für Informatik, 1989.

[RB89] Rabin, T., Ben-Or, M.: Verifiable secret sharing and multiparty protocols with honest majority. *Proceedings, 21th Annual ACM Symposium on the Theory of Computing*,1989.

[RSA78] R.L. Rivest, A. Shamir, L. Adleman, "A Method for obtaining digital signatures and public-key cryptosystems", *Communications of the ACM, vol. 21*, 1978.