# Blackmailing using Undeniable Signatures ·

Markus Jakobsson *
Department of Computer Science and Engineering
University of California, San Diego
La Jolla, CA 92093

## Abstract

With *blackmailing* we mean a situation where after a signature has been verified, the conviction of its correctness can be either kept to the verifier or, at his sole discretion, be shared with some predetermined set of cooperating co-verifiers. We show how a weakness in the protocol for undeniable signatures allows blackmailing of a signer of a undeniable signature, or several verifiers simultaneously to verify several signatures. Also, we discuss how multiple verifiers can be convinced about the correctness of a signature in similar protocols, like Designated Confirmer Signatures, although no blackmailing attack is found for here.

## 1 Introduction

An undeniable signature, invented by Chaum and van Antwerpen [1], is a signature that (A) can not be verified without the help of the signer, and (B) can not with a non-neglible probability be denied by the signer.

In [5], Desmedt and Yung disclosed a potential weakness in the protocol for undeniable signatures, in that several verifiers not trusting each other might be able to verify a signature simultaneously, without the prover of the signature being aware of proving the signature to more than one person. This could be done by setting the challenge collectively so that no true subset of the verifiers could set the challenge on their own. The attack in Desmedt and Yung's paper has certain problems, as shown in [3], but we find a way of strengthening their attack so that it will work. We also note that the protocol for undeniable signatures also has the following weakness: Alice, proving the correctness of one of her signatures, never knows what signature is being verified. We discuss how an adversary can use these two weaknesses for blackmailing a signer, Alice, who have signed some delicate message.

A similar attack will allow several verifiers to be convinced about the correctness of a signature in protocols like Designated Confirmer Signatures [2]. We will discuss how this can be done, and why it will be possible in protocols of this kind.

We will in section 2 quickly go over what an undeniable signature is and how it works - this section may be skipped by the reader who knows the protocol. In section 3 we will explain our blackmailing attack and prove that it will work as claimed. Finally, we will in section 4 discuss simultaneous verification of several undeniable signatures by several verifiers, and why all close relatives of Undeniable Signatures allows multiple verifiers to be convinced of the correctness of a signature.

## 2 The verification protocol

Assume that Alice signs a terrible secret, $m$, using the undeniable signature scheme from [1], and sends Bob $m$ and the signature, $z$. Here, $z = m^x$, where $x$ is Alice's private key and $y = g^x$ her public key, $g$ is a generator, and all operations are modulo some prime $p$.

Bob verifies the correctness of the signature by engaging in the following protocol with Alice:

$$\begin{array}{ccc} \textit{Alice} & & \textit{Bob} \\ & & a, b \in_u Z^*_{p-1} \\ & & q = z^a y^b \\ & \xleftarrow{\quad q \quad} & \\ w = q^{1/x} & & \\ & \xrightarrow{\quad w \quad} & \\ & & w \overset{?}{=} m^a g^b \end{array}$$

*The author was supported by Claes Adelskjölds fund, Gustav Sigurd Magnussons fund and von Beskows fund, The Royal Swedish Academy of Sciences

## 3 The attack

Now, say that Eve has found out $(m, z)$ in one way or another (say by corrupting/being Bob) and decides to make some money out of Alice's probable fear that her enemies will find out that she said $m$.

1. Eve informs Alice's enemies that she has some information they would like to see. She does not of course tell them what, but she asks each of them, let's call them $\{Enemy_i\}_{i=1}^n$, to generate and keep secret two numbers each, $a_i$ and $b_i$, and then calculate and keep secret $q_i = z^{a_i} y^{b_i}$ for a pair $(z, y)$ that she tells them. Next, each enemy commits to their $q_i$ to Eve. For all commitments in our attack, we will use a commitment scheme unconditionally safe for the committer [4].
2. Eve generates and keeps secret one similar pair $(a_0, b_0)$, calculates and commits to her $q_0 = z^{a_0} y^{b_0}$. After this, she sends all the participants everybody's commitments (including her own), and asks them to open up their commitments to her. The enemies open up their commitments to Eve one by one by sending her their $q_i$'s, but nobody reveals his or her $(a_i, b_i)$.
3. Eve now calculates $q = \prod_{i=0}^n q_i$. As a next step, she convinces Alice to sign some innocuous message, $m'$, whereby she gets a signature $z'$. Instead of verifying $m'$, she verifies $m$, and Alice will not be the wiser. This holds since Alice cannot tell the difference between $q$ and $q'$ as she does - as part of the protocol - not know the $(a, b)$ used for the challenge, and both $q$ and $q'$ will be uniformly distributed for uniformly chosen $(a, b)$. By participating in this verification protocol, secretly verifying $(m, z)$, Eve receives $w = m^a g^b$, where $a = \sum_{i=0}^n a_i$, $b = \sum_{i=0}^n b_i$. (Note that Eve does not know $(a, b)$.)
4. Eve will send all the enemies a commitment to $(w, m)$ and ask them to send her their $(a_i, b_i)$. When she has received and checked all the $(a_i, b_i)$'s and verified the signature, she informs Alice that she is in trouble, and if she does not do whatever Eve tells her to, Eve will execute the next step:
5. Eve broadcasts $(\{a_i\}_{i=0}^n, \{b_i\}_{i=0}^n, w, m)$. All enemies will check that all the commitments are correct, and they will all be convinced that Alice signed $m$.

We will now prove that this attack will work as it is meant to.

**Theorem 1** *If Eve does not send out* $(\{a_i\}_{i=0}^n, \{b_i\}_{i=0}^n, w, m)$ *to Alice's enemies, they have no way of verifying or even finding out the message by collaborating against Eve and Alice and eavesdropping on all communication.*

**Proof of Theorem 1:**

Assuming that the enemies have all the information they can get by sharing and eavesdropping, they know $(z, \{a_i\}_{i=1}^n, \{b_i\}_{i=1}^n, q_0, y)$ and Eve's commitment to $(w, m)$. Eve's commitment to $(w, m)$ will not help them, not even if they are computationally unlimited, as the committment scheme is unconditionally safe for the committer.

They do not have $a_0$, $b_0$ or $m$. Any two messages $m_1$, $m_2$ induce the same distribution on this known information when $a_0$, $b_0$ and the random bits for the committment scheme are chosen with a uniform random distribution, and thus they cannot calculate $(a_0, b_0)$ from $q_0$. If they do not know $a = \sum_{i=0}^n a_i$ and $b = \sum_{i=0}^n b_i$, they will not know even if Alice's proof, $w$, that she signed $m$ is correct, and they will even less be able to calculate $m$.

We note that all other information provided the enemies can be generated from $z$ in polynomial time, the commitment being independent of the actual message and $w$; $\{a_i\}_{i=1}^n$, $\{b_i\}_{i=1}^n$ being generated by the enemies, and $q_0$ being randomly and independently distributed given the rest of the information.

We also note that if given $(z, y)$ it would be easy to calculate $m$, then it would be simple to forge signatures by picking random $z$'s and computing the corresponding $m$. Although the forger couldn't verify the signature, Alice could not deny it.

■

**Theorem 2** *If Alice does not comply with Eve's request and Eve sends out* $(\{a_i\}_{i=0}^n, \{b_i\}_{i=0}^n, w, m)$ *to Alice's enemies, each one of them will be able to convince himself that Alice said $m$.*

**Proof of Theorem 2:**

Each enemy, $Enemy_i$, picks a pair $(a_i, b_i)$ at random and reveals only $q_i = z^{a_i} y^{b_i}$ and an information-theoretically secure commitment to $q_i$. Later, $Enemy_i$ receives commitments to all $a_i$'s and $b_i$'s; then $m$, $w$, $\{a\}$, $\{b\}$, satisfying $w = m^a g^b$, $a = \sum_{i=0}^n a_i$, $b = \sum_{i=0}^n b_i$. $Enemy_i$ knows that the distribution on $(w, a, b)$

depends only on $q_i$, since they were committed to before $a_i$ and $b_i$ were revealed, and the commitments are information-theoretically secure. Thus, if it were possible to falsely convince $Enemy_i$, it would be possible to falsely verify $z$ as a signature for $m$ in the original protocol.

■

## 4 Other Attacks

First, we can see that several verifiers can simultaneously verify *several* undeniable signatures by each calculating a $q_i = z_i{}^{a_i}y^{b_i}$ for possibly different signatures $z_i$. The set of enemies then send $q_i$ to Eve, the visible verifier, who calculates $q$ as the product of the $q_i$'s. After receiving $w$ and passing it on to Alice's enemies, he will require all these to open up their $q_i$ by sending him $(z_i, m_i, a_i, b_i)$. If all these were formed correctly and the signature is correct, Eve may (or may not) send out $(\{z_i\}, \{m_i\}, \{a_i\}, \{b_i\})$ to all the verifiers, after which they can verify the correctness of their signatures. Note that one verifier may trivially verify several signatures himself by using several aliases.

We note that in other related protocols, several verifiers cannot be prevented from simultaneously being convinced about the correctness of a signature, as they can jointly set the challenge so that none of them can cheat the others, and first open up their part when they receive a commitment of the answer from Alice. This will be done by the participants committing to each other to their shares of the challenge; then jointly calculating the commitment of the challenge ($q$) to be sent to Alice, without disclosing the shares to each other in the process.

## 5 Acknowledgements

I would like to thank Russell Impagliazzo for valuable discussions and support; Karin Högstedt for general support.

## 6 References

1. D. Chaum, H. van Antwerpen, "Undeniable Signatures," Crypto '89, pp. 212-216

2. D. Chaum, "Designated Confirmer Signatures", Eurocrypt '94

3. D. Chaum, "Some Weaknesses of "Weaknesses of Undeniable Signatures"," Eurocrypt '91, pp. 554-556

4. D. Chaum, C. Crépeau, I. Damgård, "Multiparty Unconditionally Secure Protocols," 20th STOC , 1988, pp. 11-19

5. Y. Desmedt, M. Yung, "Weaknesses of Undeniable Signature Schemes," Eurocrypt '91, pp. 205-220