

# Digital Signatures by Decentralized Credentials

Marius Gjerde\*    Stig Frode Mjøl̄snes†    Aslak Bakke Buan‡

October 13, 2003

## Abstract

A high security architecture employing a mobile electronic wallet was first developed in the seminal European research project CAFE. This paper deals with a generalized model based on this architecture that was recently proposed in [6], where a localized (residential) credential keeper maintains most of the content of the user's wallet, still the wallet is able to perform transactions. However, new threats against the security are introduced by this new model, hence the protocols to be used need careful construction and analysis to maintain strong multiparty security. We consider this problem here, by analyzing the new architecture of decentralized credentials, by proposing a new and efficient protocol for signature generation, and by giving arguments for the claimed security properties of this new model.

**Keywords:** Mobile commerce, e-wallet architecture, privacy, payment protocols, digital credentials.

## 1 Introduction

The seminal European research project CAFE [2] developed technology and a working pilot of the concept of an electronic wallet. The project's main concern and target were payment transactions that could replace traditional cash by digital coins [5]. The electronic wallet

---

\*Diploma thesis [3], work carried out spring 2003 at the Department of Mathematical Sciences, NTNU.

†Department of Telematics, NTNU

‡Department of Mathematical Sciences, NTNU

contained currency and credentials, and provided users the ability to perform payment transactions *offline*, ie. without contacting the bank during the payment transactions. Essential security requirements for the CAFE architecture were privacy for the user, and integrity for the service organizations. During payments, a user should be totally anonymous, and it should be impossible to link a user to a specific transaction, even if all other parties collaborated. Given several views of protocol executions, it should even be impossible to link the same (anonymous) user to two different transactions. This property is called *unlinkability*.

**Motivation** This paper covers some results from the diploma thesis work “Decentralized Credentials” [3], where the outset was the article “Online e-wallet system with decentralized credential keepers” [6] that generalizes the CAFE architecture to include *online* wallet transactions. A main idea was to separate the credentials from the electronic wallet within a fully decentralized architecture. This decentralizing requires the e-wallet to be online, as opposed to the offline e-wallet of CAFE. A major scalability challenge for most wallet schemes is the problem of multi-issuer. In practice, every service provider and credential issuer provide various types of physical tamper resistant tokens, such as smart cards, that contains and secures secret information on behalf of the service provider. It follows that users either have to juggle the cards in and out of the electronic wallet, or accommodate an ever growing number of smart cards by introducing new card slots. The introduction of a decentralized credential keeper provides a possible solution to this problem, as the keeper easily can provide a large number of card slots within its physical constraints.

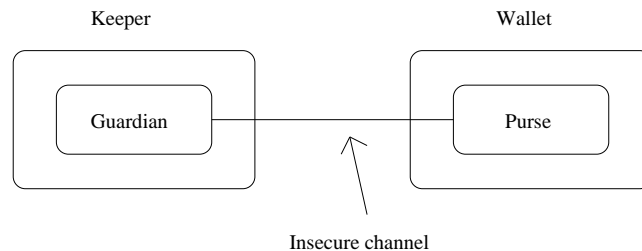


Figure 1: Segregation of guardian and wallet

The system of decentralized credentials, however, introduced some major threats against security in the architecture. In CAFE, the communication channel between the wallet and a service (ie. bank or shop) is assumed insecure, while the new system introduces an additional insecure communication channel between the wallet and its credential keeper. In turn, this introduced challenges concerning identification of the wallet associated with the keeper, and subliminal information flow [8, 9] between the credential cards at the keeper and some service.

The possible applications of the digital wallet are not limited to payment transactions [2]. It

can be used for identification, digital signatures and key agreement protocols. Some obvious applications are accessing doors and signing receipts. Other possibilities are starting cars and logging into computers.

## 2 The model

Reference [6] gives a conceptual introduction to the new architecture of decentralized credentials. Before introducing a new protocol for signature generation, we have to give a detailed description of the system's composition.

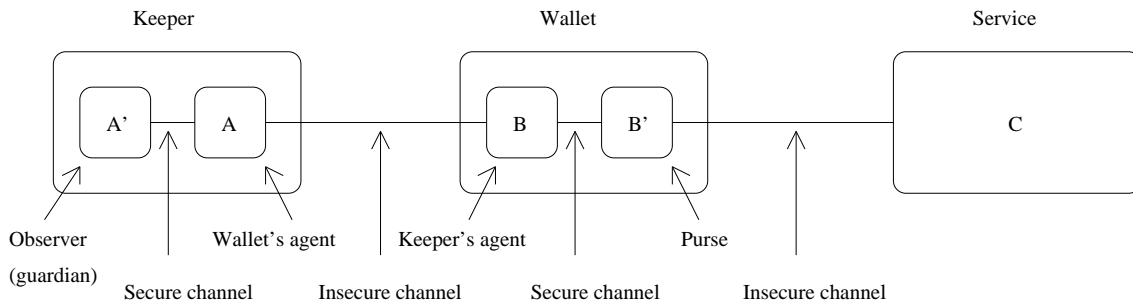


Figure 2: Decentralized credentials.

As stated, we assume the communication channels between the credential keeper and the wallet, and between the wallet and some service, to be insecure. On the other hand, the inner communication channels in the wallet and the credential keeper are assumed secure.

**Keeper** The credential keeper contains and protects hardware and software of issued credentials at some fixed location on behalf of the holder. The credential necessary to take part in CAFE transactions is called the *observer* and is issued by the bank. A special credential issued by the keeper itself, *keeper's agent* must be installed in the e-wallet to support access to the keeper and its credentials. For example, this could be a tamper-resistant smart card. The corresponding access control functionality in the keeper is denoted the *wallet's agent*. There are three approaches to establish mutual authenticity between the wallet and the credential keeper:

1. **Public key techniques** The keeper and the wallet each holds a public/private key pair.
2. **Shared secret - stored** The keeper and the wallet share a symmetric, shared key. In the wallet, the key will be stored in the keeper's agent.

3. **Shared secret - generated** Generate the secret key in the wallet by input from the user.

The passphrase approach will probably be unpractical if the security level required is similar to a standard symmetric key, because the user will have to give a very long pass-phrase as input to generate sufficiently entropy, say 64 bits. However, we note that this could avoid the need for special hardware attached to the e-wallet.

An extremely important consideration is to securely establish and maintain a one-to-one relation between the wallet and the wallet's agent, and thus exclude the ability of the wallet to contact arbitrary credential keepers to perform some action. Further work on this problem continues.

Here, we only consider the scenario of a *shared secret* stored in a tamper-proof keeper's agent, such as a smart card.

**Wallet** The keeper's agent is considered part of the the e-wallet. The main part is called the *purse* and is fully a user-controlled device that executes download protocol instructions from the different credential suppliers. An overview of the system is sketched in Figure 2. In the figure, the different parts of the system are given short names, which we will continue using to make printing easier. Thus, the observer is from now on A', the wallet's agent is A, the keeper's agent is B, the purse is B' and the service is C.

**Trust relations** The observer A' is issued by, or at least trusted by the service C. The user and the credential keeper trust the wallet's agent A and keeper's agent B, as well as the purse B', controlled by the user himself.

**Subliminal channels** A very important requirement for our protocol constructions is to preclude hidden information flow between A' and C. With the introduction of *subliminal* (covert) channels [8, 9], Simmons showed that information leakage could be present without A/B/B' being able to notice it. By this, the new protocol is constructed to preclude subliminal channels, which eliminates the possibility of information leakage from C to A' (*inflow*) and from A' to C (*outflow*).

### 3 The signature generation protocol

We now start describing the signature generation protocol for the new scheme of decentralized credentials<sup>1</sup>. The new protocol is an extension of the Schnorr signature protocol [7], and hence security is based on the complexity of calculating discrete logarithms in multiplicative cyclic groups [4].

**Setup** In the setup of the protocol, we first select two large primes  $p$  and  $q$ , such that  $q$  divides  $p - 1$ . Let  $G$  be the cyclic group  $\mathbb{Z}_q^*$  and select a generator  $g$  for  $G$ . The observer  $A'$ , or more general, a credential object  $A'$ , will hold a private key parameter  $\mathcal{S}$ , with a corresponding public key parameter  $\mathcal{P} = g^{\mathcal{S}} \bmod p$ . The parameter  $\mathcal{P}$  then represents the claimed identity (name) associated with  $A'$ , where  $\mathcal{S}$  is the trap-door information for  $A'$  to validate this claim. The public parameters  $(p, q, g, \mathcal{P})$  are assumed known to all parties in the protocol.

We next assume the wallet's agent  $A$  and the keeper's agent  $B$  share a common secret  $k$ . The size of  $k$  will correspond to the computational security complexity determined by the selected size of  $p$  and  $q$ . In addition, we define a message authentication code  $\mathcal{H}_k : \{0, 1\}^* \rightarrow \{0, 1\}^q$ , with the common secret  $k$  as input. We also define a collision-resistant hash function  $\mathcal{H} : \{0, 1\}^* \rightarrow \{0, 1\}^l$ , where  $l = 2^t$ . Detailed definitions of this kind of cryptographic functions can be found in Ref. [4].

**The protocol** In the protocol, the wallet  $B/B'$  will authenticate a message  $m$  to a service  $C$  by signing  $m$  with the help of the keeper  $A/A'$ .

The protocol is initiated by  $A$  by starting a coin-flipping protocol [4] with  $A'$ . The output of this protocol is a randomized witness  $\bar{w}$ , shared between  $A'$  and  $A$ . Before  $A$  opens the commitment  $\mathcal{H}(a)$  in the last step of the coin-flipping protocol,  $A$  sends  $\bar{w}$  as a challenge to the wallet  $B/B'$ . The wallet generates a message authentication code (MAC) on the challenge and the message, with the shared secret  $k$  between  $A$  and  $B$  as input.

It is important to include the message  $m$  in the MAC, as the message most likely have come to  $A$  from  $B/B'$  prior to protocol execution. If some adversary interfered with this prior communication, and replaced the original message  $m$  with a message  $\tilde{m}$  of his own choice, this will be revealed by adding the message  $m$  to the MAC. By this, the MAC works both to identify  $B/B'$  and to authenticate the message  $m$ .

After  $A$  has validated the identity of  $B/B'$  and the authenticity of the message  $m$ ,  $A$  opens

---

<sup>1</sup>In the diploma thesis work [3], several other protocols are constructed. These include identification, key agreement and payment in the architecture.

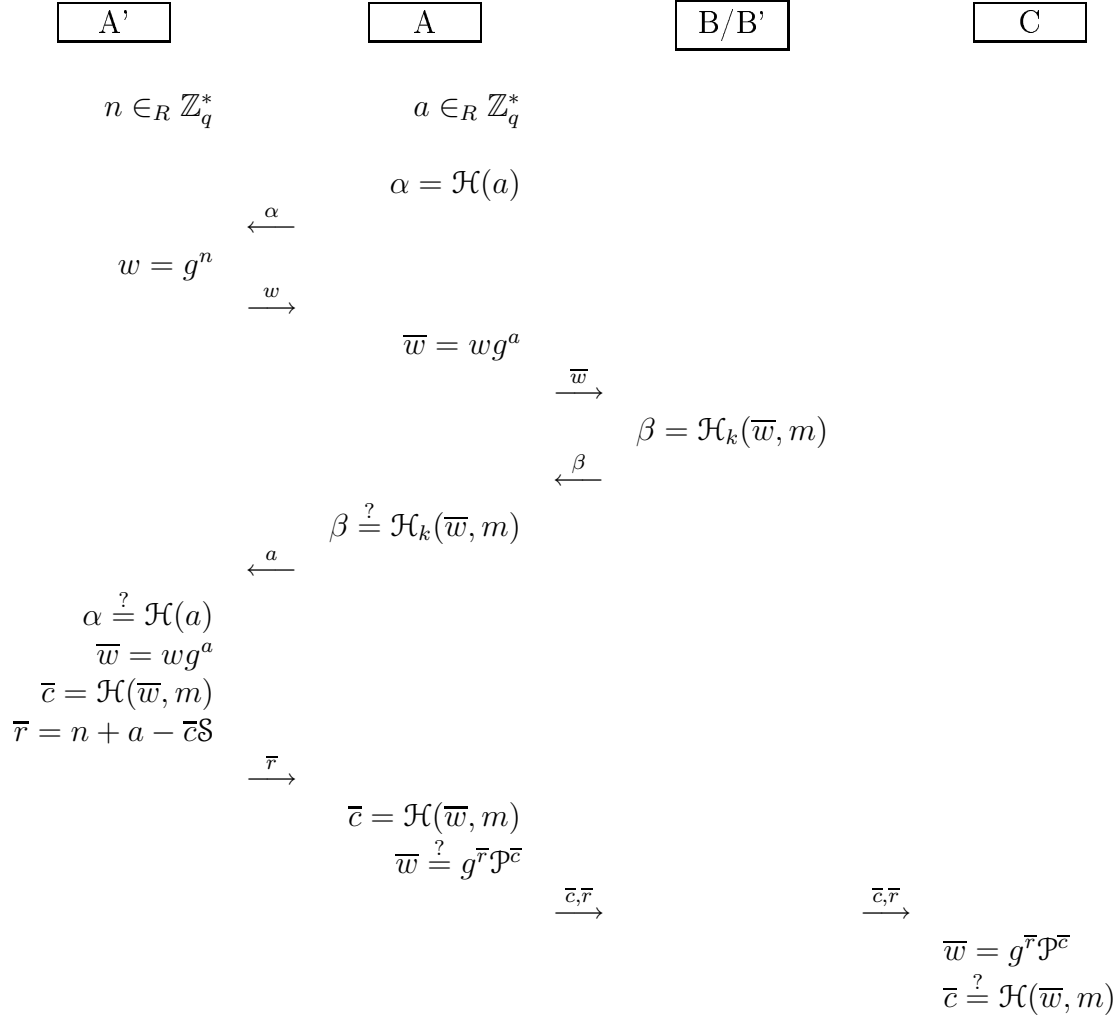


Table 1: Signature generation protocol for decentralized credentials. The identity (name) is represented by the public parameter  $\mathcal{P}$ .

the commitment  $\mathcal{H}(a)$  from the coin-flipping protocol by giving  $a$  to A'. The protocol then proceeds as the normal Schnorr signature protocol [7], with three distinctions:

1. The response from A' must be generated from the randomized witness  $\bar{w}$ , leading to the input  $a$  (from A) being included in the response.
2. A' sends only the response  $\bar{r}$  to A, and not the challenge  $\bar{c}$ , as A can compute the challenge by himself, since he knows the witness  $\bar{w}$ .
3. To preclude A' responding with an arbitrary subliminal response  $\bar{r}$ , A validates the signature before sending it to C.

However, to minimize time spent in the protocol, the validation, shown in Item 3, can be carried out off-line, after protocol execution. By this, the protocol is vulnerable to outflow, as A' can send one subliminal  $\bar{r}$  to C. This should be of little concern in protocols where the security risk by outflow is small. However, in situations where outflow has to be precluded, the validation must be performed.

The signature protocol is shown in Table 1. In the table, the tamper-proof device B and the user-controlled part B' of the wallet are put together, as they trust each other and work as one (B will contain the secret key  $k$ , which is unknown to B'). For simplicity, we omit the mod-endings from the table.

## 4 Security

We now discuss the security of the new signature generation scheme. Our protocol is an adapted version of Schnorr signature scheme. We introduce the following modifications:

1. Randomization of the witness  $w$ .
2. The introduction of the wallet B/B' as an identifying 'man-in-the-middle' with authentication of  $m$ .

We now analyze the security aspects of introducing these modifications.

**Randomization of the witness  $w$ .** We observe that the only difference between the signature generation in this protocol compared to Schnorr signature protocol [7], is the randomization of the witness  $w$  to  $\bar{w}$  by A. This randomization is done by a coin-flipping protocol [4] between A' and A, giving neither A nor A' any control of the outcome. Regardless of probability distribution of A' in choosing a value for  $n$ ,  $\bar{w}$  will be uniformly distributed if A performs according to the protocol. Thus, the signature generation of this new protocol is as secure as the signature generation in the Schnorr signature protocol.

**The introduction of the wallet B/B' as an identifying 'man-in-the-middle', with authentication of  $m$ .** To preclude unauthorized wallets being identified as A' by the service C, the wallet B/B' will have to be identified by A. In addition, the authenticity of  $m$  must be validated by A to preclude signature generation on unauthorized edited messages  $\tilde{m}$ , not originating from B/B'. In the protocol, this is done by using the ideas of the protocol SKID 2 from ISO/IEC 9798-4 [1]. The parameters used by SKID2, the shared key  $k$  and the randomized witness  $\bar{w}$ , must be sufficiently large to preclude practical attacks against this protocol.

## 5 Conclusion and Future Work

We have presented a new protocol for signature generation, adapted to the architecture of *decentralized credentials*, and argued for its security. The major advantage of the protocol is that the amount of computation done by the electronic wallet is kept to a minimum. This makes the protocol efficient, in the case where the electronic wallet has limited computing resources. The protocol also precludes inflow and outflow between the observer and the service, which gives a user the privacy of contacting whichever service he wants, without the observer getting any information about the service during protocol execution.

The security of the new signature generation scheme remains to be formally proven.

## References

- [1] ISO/IEC 9798-4. Information technology - security techniques - entity authentication - part 4: Mechanisms using a cryptographic check function. *International Organization for Standardization, Geneva, Switzerland*, 1995.
- [2] Jean-Paul Boly, Antoon Bosselaers, Ronald Cramer, Rolf Michelsen, Stig Fr. Mjølsnes, Frank Müller, Torben P. Pedersen, Birgit Pfitzmann, Peter de Rooij, Berry Schoenmakers, Matthias Schunter, Luc Vallee, and Michael Waidner. The ESPRIT project CAFE - high security digital payment systems. In *ESORICS*, pages 217–230, 1994.
- [3] Marius Gjerde. Decentralized credentials. Diploma Thesis, Dept. of Mathematical Sciences, NTNU, june 2003.
- [4] A. J. Menezes, P. C. van Oorshot, and S. A. Vanstone. *Handbook of applied cryptography*. CRC Press, 1997.
- [5] Stig F. Mjølsnes. Digital currency by cryptographic protocols. In R. Conradi, editor, *Norsk Informatikk Konferanse - NIK'89*, pages 139–157. Tapir Academic Publisher, 1989.
- [6] Stig Frode Mjølsnes and Chunming Rong. On-line e-wallet system with decentralized credential keepers. *Mobile Networks and Applications*, 8(1):87–99, 2003.
- [7] C. P. Schnorr. Efficient signature generation by smart cards. *Journal of Cryptology*, pages 161–174, 1991.
- [8] Gustavus Simmons. The prisoners problem and the subliminal channel. In *CRYPTO '83, Santa Barbara, CA*, pages 51–67, 1984.



- [9] Gustavus Simmons. The subliminal channel and digital signatures. In *Advances in Cryptology - EUROCRYPT '84*, pages 364–378, 1985.