Esecurity: secure internet & e-cash, summer 2015

MICHAEL NÜSKEN

8. Exercise sheet
Hand in solutions until Tuesday, 16 June 2015, 09:00

Exercise 8.1 (Electronic cash). (6 points)

Find one recent exposition of an electronic cash system, that is, an anonymous and account-free system for coins. (Bitcoin is not such a system.)

(i) What are the major players and top-level protocols? 2

(ii) Which primitives are used, say in black-box manner? 2

(iii) Real cash has various important properties, for example: 2

- It is difficult to forge coins or bills.
- It is anonymous: you can almost never say how the second previous owner of a particular coin was.
- It is transferable from one holder to the next several times.
- It is very difficult to copy coins (unless you are a Treckie).

What are the corresponding properties of the exposed systems?

Exercise 8.2 (Blind signatures). (8+4 points)

It is sometimes required that a signature protocol between two parties ALICE and BOB runs in such way that BOB *implicitly* signs a message m on behalf of ALICE, but does not know explicit by the message he is signing. Thus BOB cannot associate the signature to the user ALICE. Such protocols are called *blind signatures* and play a key role in electronic cash schemes and voting protocols.

We describe a blinding protocol based on the RSA signature scheme. Let BOB have the secret and public RSA keys $\mathrm{sk} = (N, d)$ and $\mathrm{pk} = (N, e)$. In order to receive blind signatures from BOB, ALICE uses her own *blinding key* $k \in \mathbb{Z}_N$ with $\gcd(k, N) = 1$.

Suppose that ALICE wants to have BOB sign the message $m \in \mathbb{Z}_N$ so that the signature can be verified but BOB cannot recover the value of m. Consider the following

Protocol.

1. ALICE sends $M = m \cdot k^e \in \mathbb{Z}_N$ to BOB.
2. BOB produces the signature $\sigma = \mathrm{sig}_{\mathrm{sk}}(M) = M^d \in \mathbb{Z}_N$ and sends it to ALICE.
3. ALICE recovers $\mathrm{sig}_{\mathrm{sk}}(m) = k^{-1} \cdot \sigma \in \mathbb{Z}_N$.

4 (i) Show that the above protocol produces a valid signature and fulfills the requirements for a blind signature scheme.

4 (ii) Now, ALICE chooses 100 messages m_i, all with the same amount but with different serial numbers, and 100 blinding keys k_i. BOB chooses j and ask ALICE to reveal all k_i with $i \neq j$. Then BOB computes a signature σ of M_j and sends it back to ALICE. Can ALICE recover a valid signature from σ for another message m'? If yes, how much control does ALICE have on the message m' (say, can she change the amount to a certain value)?

+4 (iii) Design a blind signature scheme based on ElGamal signatures and explain why it has the properties of a blinding scheme.

Hint: by using a private blinding key k, ALICE should know a function $f_k : \mathbb{Z}_p \to \mathbb{Z}_p$ with which transforms the message m into $M = f_k(m)$. With the same k he should be able to generate a function g_k such that $g_k(\text{sig}(M)) = \text{sig}(m)$, where $\text{sig}(m), \text{sig}(M)$ are, as previously, the signatures of m, M.

Exercise 8.3 (Coin flipping by telephone). (0+10 points)

(i) Read Blum (1983).

+1 (ii) What are the properties of a coin-flipping protocol? What additional properties does the proposed protocol fulfill?

+1 (iii) On which assumptions does the protocol rely?

+2 (iv) Which conditions should the modulus n satisfy? How can these conditions be checked by Alice?

+4 (v) Describe the proposed protocol and prove that the first of the properties of a coin-flipping protocol holds.

+2 (vi) How could Alice cheat if she knows a factorization of n?

Hint: Extracting square roots modulo a composite number n is computational as hard as factoring n.

References

MANUEL BLUM (1983). Coin flipping by telephone - A protocol for solving impossible problems. *SIGACT News* **15**(1), 23–27. ISSN 0163-5700. URL `http://doi.acm.org/10.1145/1008908.1008911`.