ETH Zurich, Department of Computer Science
SS 2020

Prof. Ueli Maurer
Dr. Martin Hirt
Chen-Da Liu Zhang

Cryptographic Protocols
Solution to Exercise 12

12.1 Shared Accounts and Escrow Services

a) One can consider that each account is associated with n public keys $(\mathsf{pk}_1, \ldots, \mathsf{pk}_n)$ assigned. A transaction is considered valid if it contains k signatures that are correctly verified with k corresponding public keys of the associated account.

b) Consider the scenario where A pays for a good from B. Moreover, consider an account that can be approved by 2 out of 3 parties, where the parties holding a secret key are A, B and J, respectively. Parties are instructed to proceed as follows: A transfer the payment to the shared account. Then, B sees that the transfer has been made to the shared account, it transfers the good to A. Finally, in case both parties are honest, A and B jointly transfer the payment to B. Otherwise, the complaining party notifies the judge J to resolve the corresponding dispute.

If A and B are both honest and follow the protocol, A receives the good and B receives the payment. On the other hand, if A is dishonest and does not contribute to transfer the payment to B after B transferred the good, B and J use their keys to transfer the payment to B. And if B is dishonest and does not transfer the good to A, A and J use their keys to return the payment to A.

12.2 Decoupling Parties and Users

a) A dishonest king can sign a block that was not part of the chain: when a user requests a block from a dishonest party, the corresponding dishonest king signs a potentially different tampered block, and the dishonest party sends it to the user, who will accept the block because it has a correct signature.

b) After the king broadcasts the block, all parties sign the broadcasted block and send their signature to all other parties. A block is considered *signed* if there are at least $t + 1$ signatures on the block. The rest stays the same. Note that any block that is signed is correct. However, when a user queries a party, a dishonest party can still choose not to send the block. Hence, a user needs to query at least $t + 1$ parties to ensure that he gets the correct block.