Home Assignment 1
EITN41

Anon

A-4 *What is the difference between a three-party scheme and a four-party scheme for credit card payments?*

In both schemes, two of the parties are the same: Merchant and Customers. The difference is that in four-party, the two remaining parties are Acquirer and Issuer. In a three-party scheme, Acquirer and Issuer are the same entity.

A-7 *Is SSL required in SET? Motivate your answer.*

No. Basically, SET does SSL's job plus more. SSL encrypts communication and lets a client verify the server's authenticity. In SET, Client, Merchant and the Payment Gateway has certificates and trades these, verifying authenticity of all three parts. Communication is also encrypted.

A-13 *What is the difference between* authorization *and* authentication *in VbV (3D Secure)?*

Authentication involves Client, Merchant (with Merchant Server Plug-in, MPI) and Issuer (with Access Control Server, ACS). Most oftenly it is carried out like this: after Client enters their card number (a check is performed to see if card is enrolled and if not, perform enroll process), then MPI sends Payer Authentication Request (PAReq) to ACS. ACS sends an authentication request to Client. Different authentication procedures can be used, but most often Client enters a password. The password is sent to ACS, which checks if the password is correct and with that if the authentication is successful.

Authorization is done after this, Merchant sends an authorization request to Acquirer. Acquirer sends authorization request to Issuer, which processes the request and chooses to approve or decline the authorization request.

So, in fewer words, differences between authorization and authentication are (amongst others): different agents are involved in the two processes, authentication is decided on the Client's provided credentials and authorization is decided on the Client's data with Issuer (e. g. too low balance on associated account to perform transaction).

A-23 *In PayWord, a unit could be, e.g., one cent (or one öre), so even though the payments are "micro", the hash chains could be pretty long. Could this pose a storage problem to Alice, who has to generate the entire chain when (before) she makes her first purchase from a merchant?*

The rules are, according to the article by Rivest and Shamir, for Alice to generate a hash chain before conducting business with a Vendor, and do this for every Vendor she want to buy something from. The recommendation is for Alice to cache her Vendor-specific commitment (which includes the last value

of the hash chain) until her business with Vendor is concluded, or until the commitment's expiry date is passed. If a commitment is deleted, a new one can be generated in its place.

Implications of this should be that if Alice does business with many Vendors the storage may become an issue, especially with small currency units and large costs. But, since the commitment (and with it, a new hash chain) can be computed, a trade-off can probably be done between storage space and computational cost: if Alice is low on storage space, delete old hash chains and commitments (preferably for Vendors she interact seldom with, or has not done business with in a long while) and instead use resources on computing commitments and hash chains more often.

A-27 *In the PayWord protocol, give the Bank's algorithm for verifying how much money should be taken from the user's account.*

The Bank is contacted by the Vendor/Merchant, and receives the User's commitment and the PayWord (w_i, i) with max(i) received by Vendor from User. Verification by Bank is by hashing w_i i times. If result matches w_0 (the last value in the hash chain) in the commitment, transfer i currency units from self to Vendor and demand the same amount from User.

A-29 *What is meant by a probabilistic payment? How does the Electronic Lottery Tickets scheme differ from Peppercoin from the user's perspective? How do they differ from the Merchant's perspective?*

Compare to micropayments. Instead of a User making a micropayment for μ ¤, make a macropayment for γ ¤with probability μ/γ. Pay nothing with probability $(1 - \mu/\gamma)$.

User perspective: in Electronic Lottery Tickets (ELT), there is a psychological load on the User in that they sometimes pay more than they have actually spent. In Peppercoin the User never pays more than they have actually spent and the psychological load is taken by the Bank. There is also less interactivity for User in Peppercoin.

Merchant perspective: in ELT, Merchant has to produce hash chain, in Peppercoin to provide signature (which yields a one-way interactivity). Merchant still gets the same amount of money (just surplus diff paid for by Bank instead of User).

A-31 *How much is the transaction fee for a Bitcoin transaction and how is it determined?*

The transaction fee is determined by the user that wants the transaction performed. The fee is one incentive for a miner to include transactions in their newly mined block. The fee can be nothing and upwards (fee $\geq$ 0 BTC). The higher the fee, the higher the probability that a miner will include the transaction in their block, since they can "claim" the fee as their own.

A-34 *How is the difficulty in Bitcoin block hashing adapted so that it (almost) always takes about 10 minutes for the system to produce a new block, regardless of the computational power that enters the system?*

The difficulty implementation is this: there is a number that the value of the computed hash must be less than. The difficulty is adjusted every 2016 blocks. When it is adjusted, an average is taken over the last two weeks. This is to

adjust the difficulty according to how many computers are active in the Bitcoin network.