LAB REPORT 4: DECENTRALISED DIGITAL TOKEN PAYMENT SYSTEM

Student: Gerald Kirui (1615002)

School of Electrical & Information Engineering, University of the Witwatersrand, Private Bag 3, 2050, Johannesburg, South Africa

Abstract: A decentralised digital payment system is an electronic form of payment which is not regulated by a central authority. The purpose of this report is to design the digital payment system on MATLAB® The history, overview, applications and attacks against the decentralised digital payment system, will be discussed. Moreover, the implementation details of the hash function, the signature verification function and the data structures, will be outlined. Lastly, defences the byzantine and double spending will be discussed. The results show that the time taken to mine a block with a difficulty of 1, 2 and 3 is 2.72 s, 11.85 s and 852.66 s respectively, and the digital system can validate and invalidate a transaction based on the sender's signature.

Key words: Decentralised digital payment system, SHA-256, data structures, block-mining.

1. INTRODUCTION

A decentralised digital payment system is an electronic form of payment which is not regulated by a central authority. This lab report discusses the methodology used to design the decentralised digital payment system on MATLAB®. The scope of this report encompasses the implementation details of the hash function (SHA-256), the signature-verification function (RSA) and the data structures used to store information of the decentralised digital payment system. The remainder of this document reports on the results obtained from the digital payment system, a discussion of the defence against the byzantine attack and double spending, the promotion of user privacy in the digital payment system and the problems faced in the design and implementation of the digital payment system.

2. BACKGROUND INFORMATION

2.1 History of the Digital Token Payment System

The decentralised digital payment system was developed because of the inherent flaws of traditional payment systems. The major concerns with traditional payment systems are the following: traditional payment systems are susceptible to fraud; the transaction details on a traditional cheque can be altered relatively easily. Moreover, traditional payment systems are heavily regulated by central authorities like large banks and the government; this reduces flexibility in the international economy. Lastly, because of exchange rates, weaker economies are put at a disadvantage in terms of economic growth and shipping costs. The first widely used decentralised digital token payment system is Bitcoin, designed by Satoshi Nakamoto in 2009. Since its inception, various other cryptocurrencies have been created [1].

2.2 Overview of the Digital Token Payment System

Any decentralized digital token payment system can be subdivided into the following:

- Block data structure
- Block chain data structure
- Transaction data structure
- Hash function
- Signature-verification function

Each block holds a list of transactions made between parties at different times. The block chain links two or more different blocks, like a linked list. Each transaction holds the sender's address, the recipient's address as well as a digital signature. The hash function is used for proof of work. The signature-verification function is used to validate or invalidate a transaction between two parties. If the transaction is invalid, i.e. fraud has taken place, the transaction will not be added to the block chain and therefore the transaction will not be processed [1].

2.3 Applications of the Digital Token Payment System

A digital token payment system is not regulated by a central bank or by the government. For this reason, donations and purchasing of goods and services can be done even when it is banned by the government. This allows for a free, robust economy and political climate. Such a payment system can also reduce remittances, i.e. a fee paid when sending goods abroad. This is possible because, unlike traditional currencies, a decentralized digital token payment system is not subject to strong currencies like the dollar, i.e. it is not regulated by a central bank/entity. A decentralized token payment system can be used as a form of investment. Investments are possible because the value of the cryptocurrency changes with time. Trading traditional currencies for cryptocurrencies can yield substantial monetary returns [1].

2.4 Attacks against the Digital Token Payment System

The Byzantine attack is an attack on a decentralized digital token payment system in which a malicious member

attempts to change the transaction details in a specific block. For the malicious client to successfully alter the transaction details, he has to re-calculate the hash of all the other blocks in the block chain, and broadcast the new block chain to the other members of the decentralised digital payment system. He has to do it quickly enough so that the other members can accept the new block chain as the "legitimate" one.

Double spending is an attack on a decentralised digital token payment system in which a signed transaction is duplicated. For example, Alice sends Bob R 100 and wilfully signs it with her private key. Thereafter, Bob duplicates the transaction so that he receives R 200 in total.

3. THE HASH FUNCTION: SHA-256

SHA-256 (Secure Hash Algorithm) is a hash algorithm designed by the NSA. The number of unique hashes that can be produced by SHA-256 is 2^{256}, which is an incredibly large number [2].

3.1 Implementation details

The text to be hashed is converted into binary and transformed into a multiple of 512 bits using padding. For every block of 512 bits, a 64 × 32 matrix W_t is created using rotation and the XOR-operation systematically on a group of pre-determined rows. Using matrix W_t and the k-constants (which are standard SHA-256 quantities), the initial hashes are transformed into intermediate hashes. The intermediate hashes overwrite the initial hashes, as they are fed back into SHA-256. In the last stage, the final hashes are joined to form a 256-bit quantity which form the hash of the text. Figure 1 is a high-level flow diagram of SHA-256.

3.2 Advantages and reasons for using SHA-256

SHA-256 is a very secure hashing algorithm, because the total number of unique hashes is 2^{256}. This means that SHA-256 is collision-resistant; the chance of obtaining two different messages with the same exact hash is incredibly unlikely. In the context of the cryptocurrency designed in this report, it means that transactions cannot easily be forged [2].

3.3 Limitations and disadvantages of SHA-256

SHA-256 is very time-consuming to implement. Because of its complexity, SHA-256 was also difficult to debug. Moreover, SHA-256 would result in poor performance when executing the proof of work of a list of transactions, i.e. it executed the proof of work slowly [2].

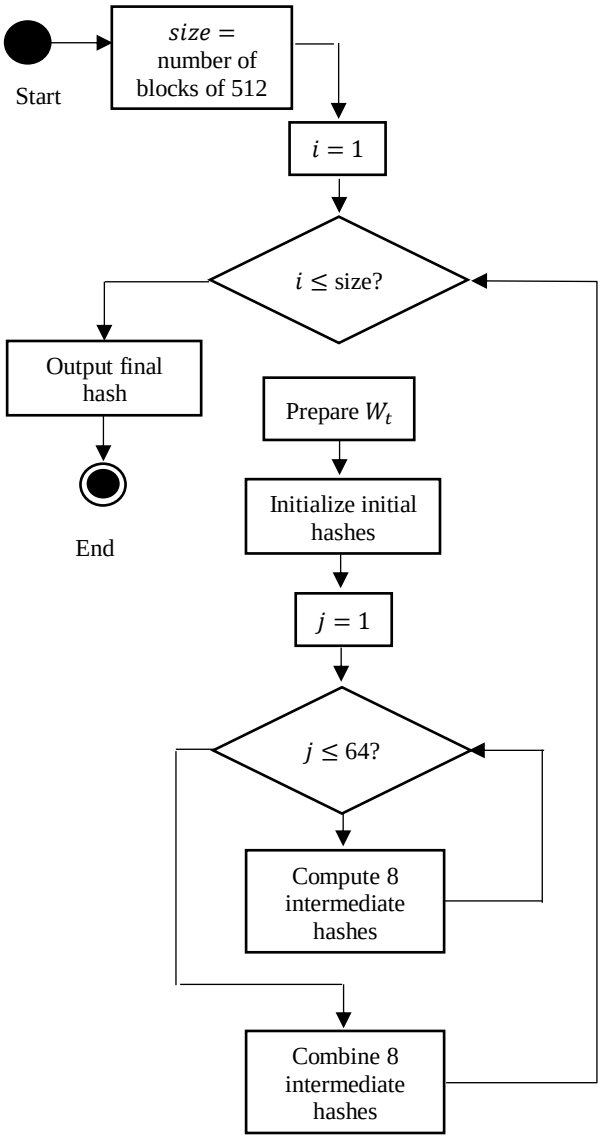

Figure 1: High-level flow diagram of SHA-256

4. SIGNATURE VERIFICATION: RSA

The RSA algorithm is an asymmetric cipher designed by MIT academics Ron Rivest, Adi Shamir and Leonard Adleman. In the context of this laboratory report, RSA is used to encrypt the hash of a transaction using the sender's private key. This action is called "signing" the transaction. To validate a transaction, the signature is decrypted using the sender's public key, and the result is compared to the hash of the transaction.

4.1 Implementation details

Two large prime numbers, p and q, were set by default to 3494053 and 3471463 respectively. The product $n = pq$ and Euler's Totient function $\varphi(n) = (p-1)(q-1)$ were then calculated. The public key e was then chosen such that e is coprime to $\varphi(n)$. This was done using the Euclidean algorithm. Lastly, the private key d, the multiplicative

inverse of e modulo $\varphi(n)$, was determined using the extended Euclidean algorithm.

4.2 Advantages and reasons for using RSA

RSA is a secure encryption algorithm. In the context of the digital token payment system, it is computationally infeasible for Bob to forge Alice's signature because *only* Alice knows her own private key, which she uses to sign the transactions. RSA was used to counter the Byzantine attack [3].

4.3 Limitations and disadvantages of RSA

In the case where many transactions are made, the RSA algorithm affects the performance of the digital token payment system because many transactions will need to be validated. It will take a substantial amount of time for transactions to be processed [3].

5. DATA STRUCTURES

Data structures are used to store important information about the clients of the digital token payment system and the transactions they make.

5.1 Implementation of the Block data structure

The Block data structure stores a list of transactions made at the same time. Therefore, a class containing the timestamp and the array of transactions as member variables, was made. To ensure that a malicious client cannot tamper with the order of the blocks, the block also contains its own hash as well as the previous block's hash as member variables. The Block data structure has, among other housekeeping functions, a function "mineBlock" which allows a client to earn rewards for mining a block.

5.2 Implementation of the Block Chain data structure

The Block Chain data structure stores a list of blocks. The blocks are arranged as a linked list, except that the blocks cannot be modified or rearranged (as would be the case in a linked list). As member variables, the Block Chain data structure has "difficulty", which determines the number of zeros that the hash of a block begins with (for proof of work), and "miningReward", which is the amount a member of the block chain earns for mining a block. Among other housekeeping functions, the Block Chain data structure has a function "minePendingTransactions" which groups a few transactions, places them in a block and joins the block to the block chain.

5.3 Implementation of the Transaction data structure

The Transaction data structure stores information about a single transaction. The member variables are the following: the sender and recipient's public key, the amount of money transferred, the time at which it was transferred as well as the sender's digital signature. It also contains two functions, "sign" and "isValid", which allows the sender to sign the transaction with his/her public key and to determine the validity of a transaction, respectively.

5.4 Advantages and reasons for using data structures

Using data structures and classes allows data to be organized, which reinforces the efficiency and simplicity of the decentralized digital token payment system.

5.5 Limitations/disadvantages of using data structures

If many objects of the data structures are created, it may affect the speed at which functions are carried out, because the objects will take up a lot of space in computer memory. Therefore, the data structures are only suitable for a local decentralised digital token payment system.

6. RESULTS

6.1 SHA-256 and proof of work

Proof of work is the act of validating the authenticity of the contents of a block before the block is added to the block chain. The difficulty in the proof of work is the number of zeros that the hash of the block must begin with. Proof of work makes it difficult for a malicious member to change the contents of a block. Changing the contents of a block will make the block chain invalid, so the malicious member will need to re-calculate the hashes of all the other blocks in the block chain, which requires a lot of computational power. To determine the how long it takes to mine a block with different difficulty, the function "minePending-Transactions" and the built in functions "tic" and "toc", were used. The graph below shows the time taken to mine a block with a difficulty of 1, 2 and 3.

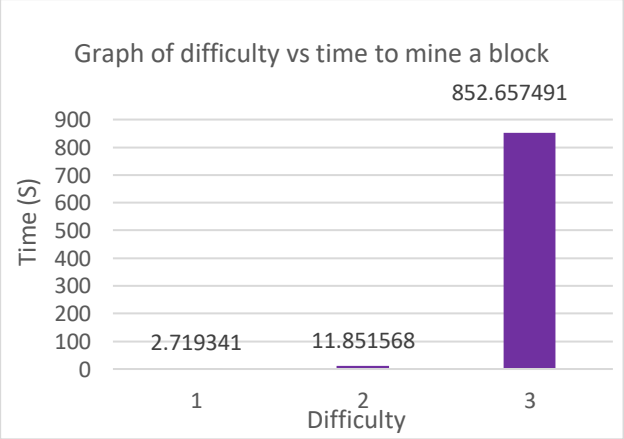

Figure 2: Graph of time taken to mine a block vs difficulty.

6.2 RSA and Signature Verification

RSA was used to generate a public key and private key for two members of the digital payment system, namely: Alice and Bob. Two transactions were made to test the signature verification of the digital payment system. First, Alice paid

R100 to Bob and signed the transaction with her own private key. The transaction is successful because Alice signed the transaction with her private key. Secondly, Bob claims that Alice paid him R50 and produces a forged signature which is not Alice's signature. The transaction is invalid because the recovered hash of the transaction does not match the original hash of the transaction. The table below shows a summary of the two transactions.

Table 1: Valid and invalid transactions

Sender	Recipient	Amount	Signature	Outcome
Alice	Bob	R100	Alice's	Valid
Alice	Bob	R50	*Forged*	Invalid

7. DISCUSSION

7.1. Defence against the Byzantine Attack

As previously stated, the Byzantine attack is an attack on a digital payment system in which a malicious member alters the contents of a block (transaction details) and broadcasts the block chain to all other members of the digital payment system. The Byzantine attack can be countered by using proof of work whenever a block chain is broadcast. Using proof of work will require the malicious member to have exceptional computational power to change the contents of an entire block chain *and* to mine each block thereafter. Since it takes a lot of computational power, the block chain produced by the malicious member will grow slower than the block chain from other honest members. Therefore, the other members will simply ignore the block chain produced by the malicious member, thereby defending the digital payment system against the Byzantine attack.

7. 2. Defence against double spending

Double spending is a form of fraud in which the same *valid* transaction is copied more than once by a malicious member in order to defraud other members. A digital payment system can be defended from double spending by using a timestamp in the transaction details. The timestamp will be compared to the time when the transaction was supposedly signed by the sender. If there is a discrepancy in the times, the transaction will be rendered invalid [4].

7. 3. User privacy

User privacy is ensured by the user's public key and private key. Instead of displaying a user's name and surname in the transaction details, the user's public key (which represents the user's account number) can be provided. This promotes anonymity, which reinforces user privacy. Moreover, the user's private key is used when transactions are made. It should be known only by the user. This means that the user's financial status is kept private from other users.

7.4. Problems in design and implementation

The main problems faced in the design and implementation of the digital payment system, are: complexity in concepts, time constraints, differences in syntax between C++ and MATLAB with regards to classes and working with no partner. To design and implement the digital payment system, information of SHA-256, RSA and object-oriented programming had to be obtained. Coding SHA-256 and the member functions on MATLAB® was time-consuming, which was problematic since only 4 weeks were given to complete the digital payment system. Lastly, the syntax in setting up a class in MATLAB® is different from the syntax of C++. This required familiarity of the syntax of MATLAB® in a short period of time. Working with no partner was problematic because all the work had to be completed with no help from a partner, which meant slow progress in the design and implementation of the digital payment system. See Appendix A for the full breakdown of the work done.

7.5. Analysis of results

As seen in the Figure 2, the time taken to mine a block with a difficulty of 1, 2 and 3 is 2.72 s, 11.85 s and 852.66 s respectively. The time taken to mine a block increases exponentially with increasing difficulty. This makes it difficult for a malicious member to change the contents of a block chain. In this manner, proof-of-work successfully defends against the byzantine attack.

As seen in table 1, the signature verification function will validate a transaction if the sender's private key is produced, but will invalidate the transaction if the signature is forged. This promotes security in the digital payment system.

8. CONCLUSION

The design of a decentralised digital payment system is successful. A digital payment system can be subdivided into a hash function, a signature verification function and data structures. Digital payment systems like Bitcoin, promote economic growth. However, they may be susceptible to attacks such as the byzantine attack and double spending. The digital payment system designed in this report shows that it takes 2.72 s, 11.85 s and 852.66 s for a block to be mined with a difficulty of 1, 2 and 3 respectively. Moreover, the digital payment method designed, can validate or invalidate a transaction based on the digital signature. The byzantine attack can be countered using proof of work and double spending can be countered using a timestamp in the transaction details. User privacy can be promoted with the use of a public key and a private key. One major problem faced in the design of the digital payment system was the time constraint.

REFERENCES

[1] Coin Desk. https://www.coindesk.com/information/what-is-bitcoin, last accessed 23/04/2019.

[2] A. Menezes, P. van Oorschot, S. Vanstone. "Handbook of Applied Cryptography", CRC Press, Inc. Boca Raton, FL, USA 1997. pp 380.

[3] W. Stallings. "Cryptography and Network Security: Principles and Practice", Pearson, 2014 pp 38, 266, 277.

[4] Sudhir Khatwani. https://www.coindesk.com/information/what-is-bitcoin, last accessed 23/04/2019.

APPENDIX A

Table 2: Table of milestones in the completion of the digital payment system.

Milestone	**Completed Over:**
SHA-256 Hash Function	1 April – 10 April
RSA	12 April – 12 April
Block Chain Class	13 April – 15 April
Block Class	15 April – 16 April
Transaction Class	16 April