YILDIRIM BEYAZIT UNIVERSITY
GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES

A COMPARATIVE STUDY OF BITCOIN AND ALTERNATIVE CRYPTOCURRENCIES

M.Sc. Thesis by
Cavidan YAKUPOĞLU

Department of Computer Engineering

May, 2016

ANKARA

A COMPARATIVE STUDY OF BITCOIN AND ALTERNATIVE CRYPTOCURRENCIES

A Thesis Submitted to

the Graduate School of Natural and Applied Sciences of Yıldırım Beyazıt University

In Partial Fulfillment of the Requirements for the Degree of Master of Science in Computer Engineering, Department of Computer Engineering

by

Cavidan YAKUPOĞLU

May, 2016

ANKARA

M.Sc. THESIS EXAMINATION RESULT FORM

We have read the thesis entitled **"** A Comparative Study of Bitcoin and Alternative Cryptocurrencies**"** completed by Cavidan Yakupoğlu under supervision of Prof. Dr. Fatih V. Çelebi and Prof. Dr. Ali Aydın Selçuk and we certify that in our opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Master of Science.

Prof.Dr. Fatih V. ÇELEBİ
(Supervisor)

Prof.Dr. Ali Aydın SELÇUK
(Co-Supervisor)

Asst.Prof.Dr. Baha ŞEN
(Jury Member)

Assoc.Prof.Dr. Şahin Emrah
(Jury Member)

Prof.Dr. Fatih V. ÇELEBİ
(Director)

Graduate School of Natural and Applied Sciences

ETHICAL DECLERATION

I have prepared this dissertation study in accordance with the Rules of Writing Thesis of Yıldırım Beyazıt University of Science and Technology Institute;

- Data I have presented in the thesis, information and documents that I obtained in the framework of academic and ethical rules,
- All information, documentation, assessment and results that I presented in accordance with scientific ethics and morals,
- I have gave references all the works that I were benefited in this dissertation by appropriate reference,
- I would not make any changes in the data that I were used,
- The work presented in this dissertation I would agree that the original,

I state, in the contrary case I declare that I accept the all rights losses that may arise against me.

ACKNOWLEDGMENTS

Foremost, I would like to express my sincere gratitude to my advisor Prof. Fatih Vehbi ÇELEBİ and my co-advisor Prof. Ali Aydın SELÇUK, for their continuous support, expert guidance, encouragement, patience, motivation, understanding and immense knowledge. Their guidance helped me in all the time of research and writing of this thesis.

Many thanks to my friends, both near and far, who always support me, encourage me and listen my problems.

Finally, I would like to thank my family for the support they provided me through my entire life. Especially, I am grateful to my mother and father to be with me spiritually even if they cannot be near me.

2016, 8 May **Cavidan YAKUPOĞLU**

A COMPARATIVE STUDY OF BITCOIN AND ALTERNATIVE CRYPTOCURRENCIES

ABSTRACT

Bitcoin is the first decentralized peer-to-peer and the most prominent cryptocurrency. Cryptocurrency is a kind of digital currency, which is built on some cryptographic algorithms, and also it is called virtual currency. Bitcoin was started to use in 2009. Since 2009, it has had great interest, investment and contribution by developers and users. It was an alternative to fiat currency, which does not have any central authority to control money. After Bitcoin successful attempt, many alternative coins have been emerged so far for different aims. Developers implemented many different versions of Bitcoin. Some of them use a big portion of Bitcoin source code like Litecoin. Some cryptocurrencies like Zcash, Ripple, Peercoin have been implemented to tackle some deficiencies of Bitcoin like privacy, scalability, energy consumption. Some were developed with innovative ideas by utilizing blockchain idea like Ethereum, Namecoin, Colored Coin. Every day, a new altcoin with different feature come to the market and it is believed that cryptocurrency will replace with fiat currency in the future.

In this thesis, Bitcoin and alternative coins called altcoins are discussed and compared according to some criteria like privacy, transaction malleability, scalability, attacks. Future perspectives are discussed and some further work is planned for this comparative study.

Keywords: Bitcoin, Zcash, Peercoin, Ripple, Dash, Ethereum, Namecoin, Colored Coin, privacy, transaction malleability, anonymity

BITCOIN VE ALTERNATIF KRİPTOPARALARIN KARŞILAŞTIRILMALI ÇALIŞMASI

ÖZET

Bitcoin ilk merkezi olmayan, eşler arası ve en çok bilenen kripto paradır. Kripto paralar bir çeşit dijital paralar olup sanal para olarak da adlandırılır.

Bitcoin 2009 da kullanılmaya başlandı ve bu tarihten itibaren büyük bir ilgi gördü. Geliştiriciler ve kullanıcılar tarafından büyük yatırımlar ve büyük katkılar sağlandı. Paranın yönetilmesi için bir merkeze bağlı olan günümüzde kullanılan paraya alternatif olarak kullanıma sunuldu. Fakat bu süre zarfında Bitcoin birçok problemle karşılaştı. Bunlara rağmen yaklaşık yedi senede çok büyük bir pazara sahip oldu. Bitcoin'in başarılı çıkışından sonra farklı amaçlara hizmet eden bir çok alternatif kripto para geliştirildi. Geliştiriciler Bitcoin'in bir çok farklı versiyonlarını geliştirdi. Bunlardan bazıları, Litecoin gibi Bitcoin'in kaynak kodunu kullanarak geliştirildi. Bazı kripto paralar Zcash, Ripple, Peercoin Bitcoin'in mahremiyet, ölçeklenebilirlik, enerji tüketimi gibi farklı eksikliklerinin üstesinden gelmek için geliştirildi. Bazıları ise, Ethereum, Namecoin, Colored Coin gibi, blockchain denen herkese açık kayıt sistemi fikrini kullanarak yenilikçi fikirler getirdi. Her geçen gün yeni kripto paralar kullanıma hazır hala getiriliyor ve bu teknoloji gelecekte şu anda kullanılan paranın yerini alacağı tahmin ediliyor.

Bu tezde, Bitcoin ve alternatif kripto paralar tartışıldı ve mahremiyet, ölçeklenebilirlik, ticari işlem değiştirilebilirliği ve şu ana kadar maruz kaldıkları, savunmasız oldukları ataklar açısından karşılaştırıldılar. Gelecekteki durumları tartışılarak, gelecekte yapılacak işlerin planı yapılarak bu çalışma tamamlanmış oldu.

Anahtar Kelimeler: Bitcoin, Zcash, Dash, Ripple, Peercoin, Ethereum, Namecoin, Colored Coin, mahremiyet, anonimlik, ticari işlem değiştirilebilirliği

CONTENTS

ABBREVIATIONS

ASIC	Application Specific Integrated Circuit
BIP	Bitcoin Improvement Proposal
BTC	One unit of value in Bitcoin
Bitcoin-NG	Bitcoin Next Generation
CPU	Central Processing Unit
DNS	Domain Name System
DoS	Denial of Service
DSA	Digital Signature Algorithm
ECDSA	Elliptic Curve Digital Signature Algorithm
FPGA	Field-Programmable Gate Array
Ghash	Giga hash
GPU	Graphics Processing Unit
ICANN	Internet Corporation for Assigned Names and Numbers
IRC	Internet Relay Chat
I2P	The Invisible Internet Project
Mt.Gox	Magic: The Gathering Online eXchange
PoW	Proof of Work
PoS	Proof of Stake
P2P	Peer-to-Peer

RIPEMD	RACE Integrity Primitives Evaluation Message Digest
RTXP	Ripple Transaction Protocol
SHA	Secure Hashing Algorithm
SMTP	Simple Mail Transfer Protocol
SPV	Simplified Payment Verification
TCP	Transmission Control Protocol
TLD	Top level domain
Tor	The Onion Routing
TXID	Transaction identity
TPS	Transaction per second
TWh	Terawatt hour
XRP	One unit of value in Ripple
zk-SNARKs	Zero Knowledge proof which use Succinct Non-interactive ARguments of Knowledge

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

Cryptocurrency is a kind of money, which is based on cryptography to rule transactions, and creation of new money units in a secure way. Cryptocurrency is a virtual currency that is not similar fiat currency. Fiat currency is produced and ruled by a governmental law [1] and based on a trust on a third party [2].

Bitcoin is the first successful decentralized cryptocurrency attempted by Satoshi Nakamoto in 2008 with the paper "Bitcoin: A peer-to-peer electronic cash system" [3]. Bitcoin is started to launch in January 2009. Since it was launched, it has become very popular in trade and it becomes the most expensive and valuable money with 1 Bitcoin = $ 420 at the time of writing. The most valuable fiat currency is Kuwaiti dinar with 1 KWD =$ 3.31 USD with a much less valuable than Bitcoin. It is clear that Bitcoin and its derivatives will lead currency world in the future. Because it has low transaction fees and anonymous transactions with some extensions, it became a popular currency in the market and it has a promising future according to present Bitcoin data. Nowadays, Bitcoin has nearly 220,000 transactions per day[1]. Bitcoin has a high capitalization in the market with $ 6,459,306,122 USD[2] which is greater than two-fold market capitalization of Jamaica in 2014 [4]. Bitcoin reached $ 1200 USD in December 2013 and it is traded nearly at this value for a while. Even though, its price dropped from $1200 USD, it is still the most valuable currency in the world including fiat currency. It opened a new page for trades with cryptocurrency and it seems like that it will change the world currency equations.

Along with Bitcoin, new alternative currencies have been emerged with the name *altcoins*. They are similar to Bitcoin in terms of cryptographic background but they bring some innovative ideas to Bitcoin. While some of them are proposed with new developments like Zcash [5,6] for privacy problem, Peercoin [7] for energy

[1] For more information: https://blockchain.info/charts/n-transactions?timespan=1year

[2] For more information: https://coinmarketcap.com/ 09.04.2016

efficiency problem and scalability problem, Ripple [8] for network and exchange problem, Ethereum [9] uses Bitcoin as a base idea for the new innovation called programmable blockchain. Until December 2015, 2681 altcoins have been created [10]. Some of them have big market capitalization like Bitcoin. Ethereum is the second altcoin has the largest market capitalization with $ 767,326,355 in the market. Ripple follows Ethereum with $ 221,787,944 USD[3]. Market capitalization of all altcoins and Bitcoin is $ 7,982,906,688 and Bitcoin consists of nearly 80% of market capitalization of all cryptocurrency market. In this thesis, we aim to compare Bitcoin and alternative cryptocurrencies in terms of privacy, scalability, transaction malleability and attacks to understand which technical property is useful for these aspects.

In first chapter, technical background for cryptocurrency and related work are explained. In the second chapter, some altcoins are detailed in technically. In the third chapter, Bitcoin is explained in technical detail. In the fourth chapter, Bitcoin and some major altcoins like PeerCoin, Zcash, Ripple and Dash are compared according to some criteria mentioned above. In the fifth chapter, other different altcoins are explained in detail in comparison to Bitcoin. In the sixth section, there is a discussion and conclusion about the thesis in an extensive look.

1.1 Background for Cryptocurrency

In this section, there are explanations of common structures of cryptocurrency system.

1.1.1 Cryptographic Hash Functions

Cryptographic hash functions are used to convert an arbitrary sized data to fixed sized arbitrary data with unlinkability between plain text and output. Output is called hash value, hash code or hash. Hashes are one-way functions that plain text cannot be recoverable from hash values. In general, hash functions are used for integrity purpose. For example, Hash-Based Message Authentication Code (HMAC) is a specific hash usage area that checks message authentication.

---

[3] For more information: https://coinmarketcap.com/

Cryptocurrencies are using nearly 91 different hash functions at the time of writing[4] [10]. SHA2-256 and Scrypt hash functions have by far the highest usage percentage in all coins as shown in figure 1.1.

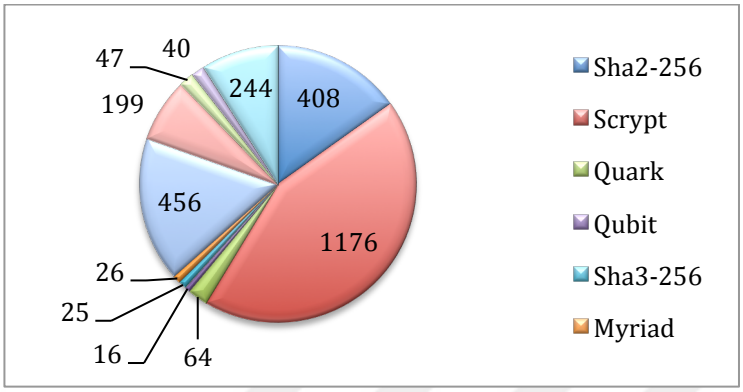

Figure 1.1 Hashing algorithm distribution in all cryptocurrencies [10].

Hash functions must have these three properties.

1. *Pre-image resistance:* For given hash value, h, it cannot be found m which satisfies hash(m)=h. This is about one-way function property of has#h functions. If a hash function does not have this property, it is vulnerable to preimage attacks [11]. For an *n*-bit hash, this attack has a time complexity 2^n.
2. *Second pre-image resistance:* For given hash of m1, it cannot be found another m2 that satisfies hash(m1)=hash(m2) If a hash function does not have this property, it is vulnerable to second preimage attacks [11]. For an *n*-bit hash, this attack has a time complexity 2^n.
3. *Collision resistance:* It is not possible to find two different m1 and m2 which satisfies hash(m1)=hash(m2) [11]. If it is possible, it is called *hash collision*. For an *n*-bit hash, this attack has a time complexity $2^{n/2}$. Unless the hash value is not at least twice as long as that required for preimage-resistance, some collisions can be found by a birthday attack [12].

[4] Date: 05.02.2016

SHA-256 is one of the most popular cryptographic hash algorithm in cryptocurrencies used in Bitcoin. Also, Scrypt, X11 are widely used hash functions in cryptocurrency world.

1.1.2 Merkle Trees

Ralph Merkle proposed Merkle tree to verify data integration efficiently which uses binary hash trees [13,14]. There are data in leaves to check their integrity. Parent nodes are hashes of concatenation of two child nodes. This calculation continues towards root node. The final hash of whole tree is the hash of the root. Figure 1.2 shows how a merkle tree to be constructed.

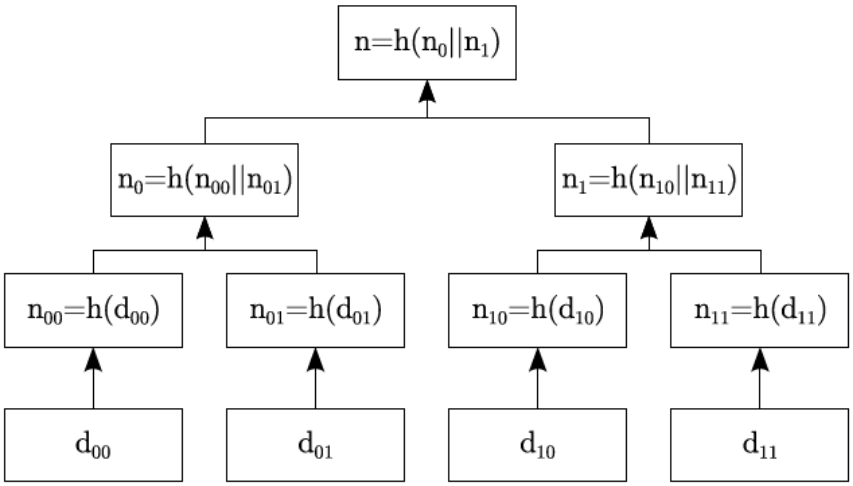

Figure 1.2. Merkle hash tree [15].

In Bitcoin, every transaction information is hashed and stored in leaves and a root hash value is created from leaves towards the root. The hash value of root represents all transactions in the tree and it is used to check integrity of transactions in the tree. When a transaction is changed, the root value is changed automatically. So, integrity of all transactions is checked easily.

1.1.3 Digital Signatures

Digital signatures are kind of mathematical mechanism to prove ownership of data who has signature keys. Digital signatures are used with the aim of authentication, non-repudiation and integrity.

In digital signatures, asymmetric cryptography is utilized. Asymmetric cryptography consists of two different keys which one is public key known by everyone, which one is private key only known by its owner. Public and private keys generated by an algorithm and private key cannot be derived from public key by others. In digital signatures, user signs his data with his private key. Public verify him as the owner of the data by using his public key. This property is represented as authentication. Also, user cannot claim that data is not signed by him. This is called as non-repudiation. After user signing the data, he sends the signed data another one. The receiver checks with public key the signed data if it is altered during transfer. If receiver validates the sender's signature with sender's public key, it means that the data is not altered during transmission. This property is named as integrity.

A signature algorithm scheme must have two properties. First one, there must be a public-private key pair to verify the signature signed by private key with the corresponding public key. The second one, it must be computationally infeasible to find the public key to sign a data in the name of another one.

Digital signature schemes have generally three algorithms.

1. **Key Generation Algorithms:** This algorithm selects a private key from a set of private keys. This algorithm gives private and corresponding public key.
2. **Signing Algorithm:** The data is signed with given private key.
3. **Signature Verification Algorithm:** This algorithm verifies the owner of data with public key of the data owner with a related technique with signing algorithm.

1.1.4 Zero-Knowledge Proof

Zero-knowledge proof is a kind of method consists of two parties called prover and verifier. Prover proves to verifier that a statement is true without revealing any data about statement except its being true. Goldwasser et al. defined zero-knowledge proof in 1985 [16]. If prover proves the possession of secret information, this is a type of zero-knowledge proof called zero-knowledge proof of knowledge. For easier understanding, Quisquierter et al. explain it with a basic scenario, by using a prover

called Peggy, a verifier called Victor and a cave example [17]. There is a ring-shaped cave and there is an entrance one side and a magic door in the cave on opposite of the entrance. Peggy claims that she knows the secret word to open the secret door and Victor wants her to prove it. And Peggy does not want to say the secret word to Victor, but she wants to prove it to Victor. They label the paths as A and B. So, Peggy enters cave from one of the two paths but Victor does not know this path. After Peggy enters cave, Victor enters cave and shouts the path name that he wants her to use while she comes to the entrance. If Peggy knows the secret word, she can open the magic door if it is necessary and come from the path that Victor wants. If Peggy does not know the secret word, she has 50% chance to come from the path that Victor wants. Because Victor chooses paths randomly to understand if she knows the secret word. So, she has 50% chance to find way correctly. If she does not know the secret word, they repeat this 20 times and they her chance to come the right path decreases to zero (nearly one in 1.05 million). So, Victor can be sure that Peggy knows the secret word after many try.

A zero-knowledge proof satisfies three properties:

- **Completeness:** If the statement is true, the honest verifier is convinced by the honest prover.
- **Soundness:** If the statement is false, dishonest prover cannot convince the honest verifier except with small probability.
- **Zero-Knowledge:** If the statement is true, dishonest verifier cannot learn anything else about the statement.

1.1.5 Blind Signature

David Chaum proposed blind signatures in 1983 to sign data by blinding it [18]. It is used in digital cash systems and electronic voting systems. It is utilized when message author and signer are different parties.

Blind signature is implemented with public key cryptography like RSA. To perform blind signature, message must be blinded with a random blinding factor firstly, then

signer signs the blinded message. This signed message can be verified publicly with blinding factor and corresponding public key.

In electronic voting systems, blind signature is used to provide integrity of election. When each ballot is verified by election authority to understand owner of ballot is allowed to vote or not and owning only one ballot by voter. In this process, election authority must not see the voter's choice and authority must sign if the ballot is valid. So, blind signatures are essential in this kind of process.

1.2 Related Works

David Chaum proposed untraceable payments with blindly signed coins in 1983 including a bank system [18]. Blind signatures construct a system that bank cannot find a link between coin and its owner. It provides unlinkablity to banking system. In 1990, Chaum et al. proposed a system that removes bank from the payment phase [19]. Camenisch et al. construct efficient off-line anonymous e-cash schemes [20]. Rivest and Shamir proposed "PayWord" and "MicroMint" to mint small micropayments over the Internet by using "Milliecent" scheme [21, 22]. Lots of electronic cash systems are tried like DigiCash in 1998 [23]. Also, Goldschlag et al. proposed publicly verifiable lotteries in 1998 [24].

In 2004, Rivest developed Peppercoin [25] to use electronic cash in practical life but it failed like Digicash. "Proof-of-Work" aka POW was first proposed in 1999 in the paper by Jakobsson and Juels [26]. Proof-of-work system is used to combat junk mails by Dwork and Naor [27]. Back offered HashCash in 1997 as a proof-of-system to detect email spam and denial of service attacks [28, 29] and recently in Bitcoin it is used as a mining algorithm. In the paper with title “Auditable, Anonymous Electronic Cash”, a system is offered to maintain the integrity of a public database instead of bank like public ledger in Bitcoin to detect double spending [30].

Dai proposed B-Money in 1998, which is the first anonymous transaction system open to public [31]. Smart contract term was used firstly by Szabo in 1997 that indicates that “Smart contracts combine protocols with user interfaces to formalize

and secure relationships over computer networks*"* [32]. Between 1998 and 2005 Szabo created *Bit Gold* [33].

Bitcoin was proposed with a whitepaper with the title "Bitcoin: A Peer-to-Peer Electronic Cash System" by a pseudonym person or group of people with nickname Satoshi Nakamoto in 2008 [3]. The *genesis block,* the first block of Bitcoin system, was mined in January 3, 2009 by Satoshi Nakamoto [34]. With time, Bitcoin became prevalent as cryptographic money in the market and price of one unit of Bitcoin currency risen up to $1200USD in 2013. But after that time its price decreased to $372USD in December 2015. Bitcoin faced lots of problems like theft [35] with the biggest one in MtGox [36], ransomware like Cryptolocker [37] and association with crime usage in Silkroad [38].

After Bitcoin, lots of alternative coins have been attempted like Litecoin [39], PeerCoin [7], Primecoin [40], NameCoin [41], Dash [42], ZeroCash [5], Ethereum [9]. Ripple called XRP [43]. Also, blockchain is used as a service called Ethereum Blockchain as a Service (EthBaaS) on Azure [44]. Colored coin is proposed by Rosenfeld to provide exchange of many kinds of assets by using blockchain idea [45]. Colored coins can represent different kinds of assets like house, car, stocks, bonds as "tokens" to transfer of ownership of these assets in a fast, transparent and low-cost way.

At the time of writing, in December 2015, there are 2681 altcoins in total [10]. 620 altcoins are mineable at the time of writing. Peercoin is the first cryptocurrency which uses Proof-of-Stake (PoS) [7]. PoS is an alternative mining system to Proof-of-Work system used in Bitcoin. PoS is a kind of mining method that reduces energy consumption of miner with the coin age idea. Another alternative to PoW system is Proof-of-Activity (PoA) is proposed by Bentov et al which promises low energy usage, less vulnerable to double spending attacks and better network topology [46]. CryptoNote was proposed as an application layer protocol for decentralized, private digital currencies by preventing learning sender and reciever of a coin [47, 48]. It provides untraceable payments, unlinkable transactions, double spending proof. Bytecoin (BCN), Monero (XMR), Aeon (AEON), DigitalNote (XDN), Boolberry (BBR), DarkNetCoin (DNC), Quazarcoin (QCN), Fantomcoin (FCN), Moneta Verde

(MCN), Dashcoin (DSH), RedWind (RD), Breakoutcoin (BRO), CryptoNoteCoin (CNC) are CryptoNote currency attempts.

CHAPTER 2

BITCOIN

Bitcoin is the first successful cryptocurrency attempt launched in 2009 by Satoshi Nakamoto. It is a kind of virtual currency which depends on *Nakamoto consensus*. Nakamoto consensus offers a decentralized, pseudonymous system, which is the core of its success [49]. Bitcoin currency code is BTC and currency symbol is ₿ (B with two vertical lines through it) created by Satoshi Nakamato. Bitcoin currency unit is *bitcoin* (with lowercase b) which is 100 million satoshi. Satoshi is the smallest unit of bitcoin currency.

Bitcoin relies on blockchain idea. Blockchain is the public ledger storing all transaction on it and chain of blocks, which are tied up each other with cryptographic algorithms. The first block is called genesis block [34] generated by Satoshi Nakamoto.

In Bitcoin, coin creation is defined as *mining* by *miners* which depends on *Proof-of-Work.* Proof-of-work is kind of cryptographic calculation, which uses hashing algorithms like SHA256 [50], Scrypt [51], Ripemd160 [52] to find a suitable hash value. Miners get rewards as coins called "bitcoin" currency name of Bitcoin system supplied by Bitcoin system to miners in case of finding appropriate hash output. So, miners use a lot of power to find the desired output at the desired difficulty level, which is fixed Bitcoin system. Transaction between users is validated by miners and miners get transaction fees from these users. These transaction fees and reward for miners are incentives for Bitcoin system.

The system runs on a big scale network using the blockchain. Bitcoin depends on ledger specifically named blockchain which is a distributed transaction database system held by independent peers on the network. Blockchain is secured by cryptographic algorithms and structures. Basically, users sends their virtual money to others and these transactions are stored in blockchain under some cryptographic assurance to prevent double spending and assurance of money of user.

Miners mine new blocks tied to previous and this rule goes to the genesis block for getting reward as bitcoin in Bitcoin system while verifying transactions published by users on the peer-to-peer network. When miner finds a block satisfies the target hash with the certain difficulty decided by system, they get reward as bitcoin. Mining reward is started with 50 bitcoins a.k.a 50 BTC. This reward is called as incentive of Bitcoin system. In Bitcoin paper, it is written that Bitcoin has incentive as reward and transaction fees [3] for system durability. While some researchers claims that is not incentive-compatible [53], some researchers accept that Bitcoin is incentive-compatible [54,55].

Bitcoin mining reward is halved in every 4 years. At the time of writing, reward is 25 BTC. At the end, there will be 21 million bitcoins in Bitcoin system. This number will be reached in 2140 and mining will end at this date. After that time, only transaction fees will run the system without reward as incentive. In Bitcoin, low transaction fee is one of the reasons of its popularity. But some research claims that low transactions fees cannot continue in Bitcoin network in the future [56]. Because mining reward will decrease with time and miners need to get higher transaction fee.

2.1 Blocks

Block is a kind of data structure, which stores transaction logs in it and mined by miners. Blockchain consists of a series of blocks connected to each other with cryptographic algorithms. Block structure consists of two parts; block header and payload. Block header contains nVersion, HashPrevBlock, HashMerkleRoot, nTime, nTime, nBits, nNonce. Payload contains all transaction data. SPV (Simplified Payment Verification) clients store only block headers but full clients downloads whole block data. $\mathrm{SHA256}^2$ stands for SHA256(SHA256(data)). Table 2.1 shows Bitcoin block fields, their sizes and descriptions.

Table 2.1 Block structure.

Field Name	Type(Size)	Description
nVersion	int (4 bytes)	Block format version (currently 4)
hPrevBlock	uint256 (32 bytes)	Hash of previous Block Header $\text{SHA256}^2(nVersion\|\|..\|\|nNonce$
HashMerkleRoot	uint256 (32 bytes)	Root hash of the Merkle tree constructed from all transactions
nTime	unsigned int (4 bytes)	Timestamp in UNIX format of block creation time
nBits	unsigned int (4 bytes)	Target difficulty for proof-of-work
nNonce	unsigned int (4 bytes)	Value which gives the target hash
#vtx	VarInt (1-9 bytes)	Number of transactions in vtx
vtx[]	Transaction (Variable)	Vector of transactions

Blocks are connected each other via hash of previous block header with hash of concatenation of block header data as in below. Block header is 80 bytes.

SHA256^2(nVersion || HashPrevBlock|| HashMerkleRoot || nTime || nBits || nNonce)

Figure 2.1. shows how to connect two consecutive blocks with double SHA256.

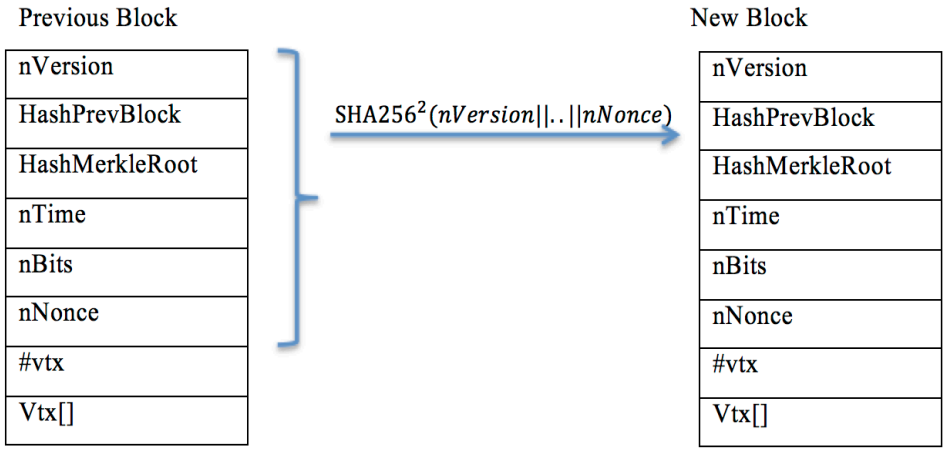

Figure 2.1 Block connection representation [15].

2.2 Blockchain

Blockchain is a chain of blocks. Blocks store block header and transactions. Blockchain is public ledger which has records all transactions done till now. Every full node downloads all blockchain to validate future transaction with checking previous transactions to control validity of bitcoins that user has. When new block is found referencing the last block of main chain, it is added to blockchain. So, blocks lie in chronological order in blockchain.

Every miner can find a different block that satisfies necessary conditions. In this case, blockchain has forks. The decision which fork will continue as main chain depends on the *longest chain* idea. For miners, the longest chain is accepted as main chain and they continue mining from that branch. Other branch blocks remain as orphan. Figure 2.2 shows blockchain flow with forks and orphan blocks.

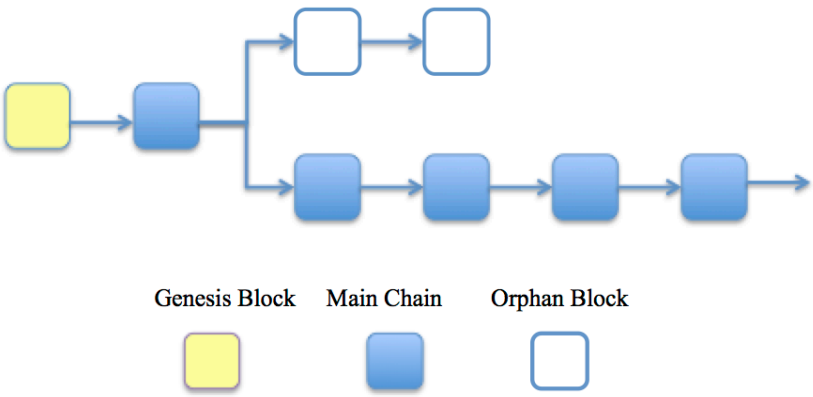

Figure 2.2 Blockchain represantation.

Block height is the number of blocks from the genesis block. Height of genesis block is 0. All blocks have a pointer to show previous block with its hash except genesis block. Genesis block stores *hashPrevBlock* as zeros string with 64 string length.

2.3 Digital Signature Algorithm: ECDSA

In Bitcoin, Elliptic Curve Digital Signature Algorithm (ECDSA) is used to sign transactions and create addresses for users to ensure that coins can be spent by only their owners by publishing transactions on public ledger (blockchain).

Every Bitcoin address is a cryptographic hash of an ECDSA public key. Bitcoin transactions are sent from and to electronic bitcoin wallets, and are digitally signed with ECDSA for security of bitcoins of users. Bitcoins are only records written in incoming transactions to users. User can spend only bitcoins in incoming transactions.

ECDSA is a kind of Digital Signature Algorithm (DSA) that uses elliptic curve cryptography.

In Bitcoin,

- Private Key is a randomly generated number used to sign transaction. It is single unsigned 256-bit integer.
- Public Key is computed from the private key. Public key is used to verify the signed data if it is signed by the owner of the public key and transferred successfully. Compressed public keys are 33 bytes.
- Signature is a mathematical scheme produced with private key from hash of data. Signatures can be 73, 72, or 71 bytes long [57].

In comparison to DSA, ECDSA has smaller key size than DSA. In the special publication of NIST, it is claimed that for 80 bits of security, DSA needs 1024 bits public key length, 160 bits private key length but ECDSA needs 160-223 bits range key size [58]. This means that ECDSA is faster than DSA at the same security level [59]. On the other hand, Vaudenay claims that for many cryptographic schemes, if DSA and ECDSA are used in a poor way, they are highly vulnerable [60].

In Bitcoin, user's bitcoins are in wallet under security of ECDSA private keys. So, bitcoins of user are secure as ECDSA key. If user's private key is stolen, it means that user's bitcoins are stolen.

If there is not an ECDSA threshold scheme in Bitcoin system, bitcoins are subject to a single point of failure. In addition to ECDSA, Gennaro et al. presented an efficient and optimal scheme that provides a threshold DSA algorithm and an application to secure Bitcoin wallets [61]. Till then, there was not any optimal threshold DSA algorithm because of its difficulty.

2.4 Addresses

In Bitcoin, users have addresses to send bitcoins each other. Two kinds of addresses are derived from public key with two different algorithms which one is Pay-to-PubkeyHash Address, which one is Pay-to-ScriptHash Address. Public key is derived from private key by using ECDSA algorithm. Bitcoin addresses can be created offline. Also, Bitcoin addresses are case-sensitive 26-35 alphanumeric characters starting with 1 or 3 with the exception that the uppercase letter "O", uppercase letter "I", lowercase letter "l", and the number "0" are never used to prevent visual ambiguity supplied by BASE58 Encoding. But most Bitcoin addresses are 34 characters [62]. In creating process of addresses, SHA256 and RIPEMD160 [52] hash functions, BASE58 Encoder are used starting from public key for PubkeyHash Address, redemption script for Pay-to-ScriptHash Address. Bitcoin addresses and public keys are known by public to check validity of user and bitcoins of user.

Addresses are used to send bitcoins between users via Bitcoin network. Every user has an address or more than one address. Address is a kind of pseudoanonymous name for user. Some user prefers to publish his address to public, while others send the concerned user. So, a published addresses owned by a real person allows monitoring his transaction activity by everyone. This property of blockchain invades privacy of user.

Every address must be unique. The probability finding same address is nearly zero. Because probability of colluding of two Bitcoin addresses is $1/2^{160}$.

There are two types of addresses in Bitcoin called as Pay-to-PubkeyHash Address and Pay-to-ScriptHash Address.

A pay-to-pubkeyHash address is computed from the public key generated by ECDSA. A Pay-to-ScriptHash Address is computed from redemption script. Figure 2.3 shows the P2PKH and P2SH addresses creation process.

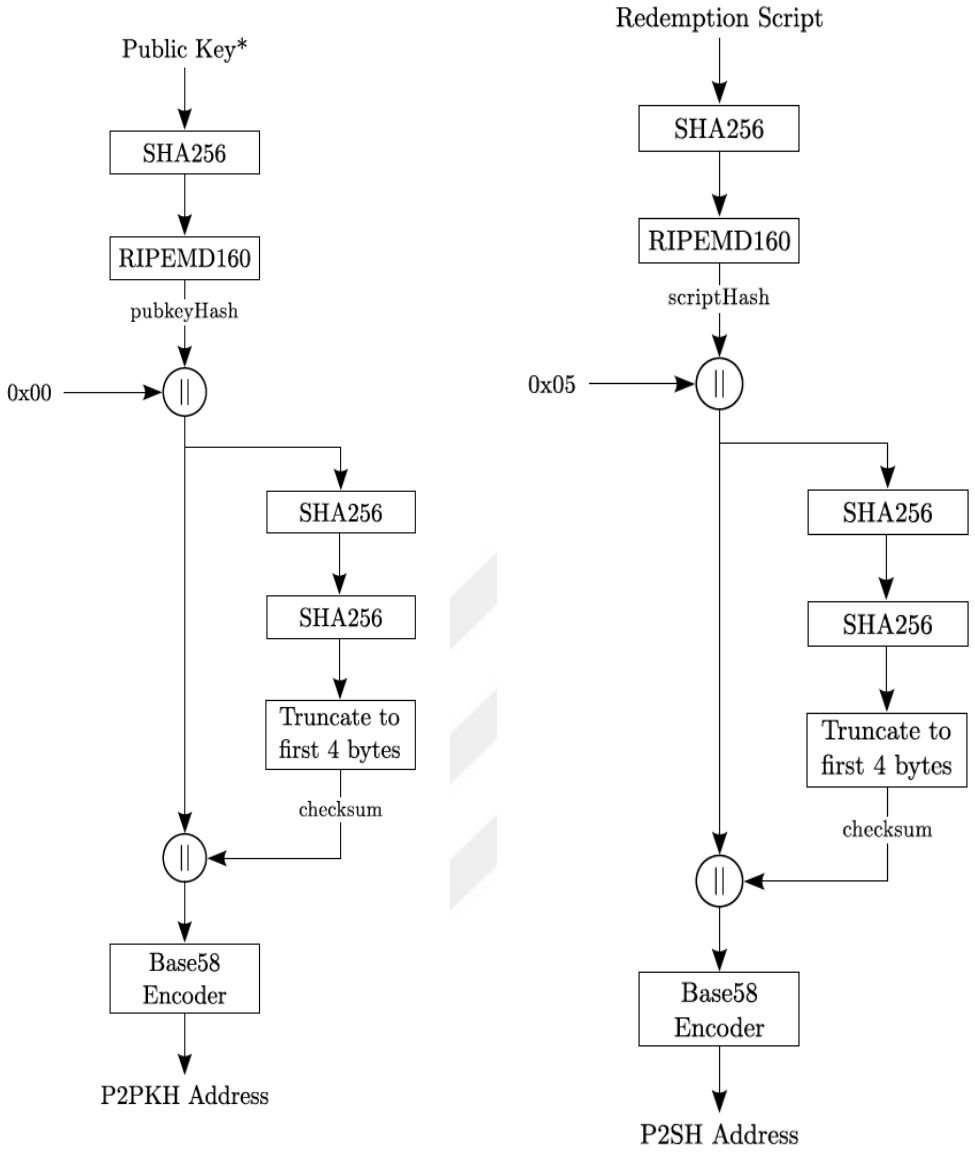

Figure 2.3 Bitcoin address conversions.

2.5 Transaction

There are two types of transactions in Bitcoin called *coinbase* and *regular* transaction. Coinbase transaction is defined as bitcoins coming from mining. Regular transaction is transfer of bitcoins between users like merchants and customers. Coinbase transaction is a specific type of regular transaction. While in regular transaction input vector size can be more than 1, in coinbase transaction input vector size (#vin) is always 1. Table 2.2 shows a transaction data fields and their descriptions.

Table 2.2 Regular transaction structure [15].

Field Name		**Description**
nVersion		Transaction format version(Currently 4).
#vin		Number of transaction inputs in vin.
	hash	Double-SHA256 hash of past transaction.
	n	Index of a transaction output within the transaction specified.
	scriptSigLen	Length of scriptSig field in bytes.
	scriptSig	Script to satisfy spending condition of the transaction output (hash, n)
	nSequence	Transaction input sequence number.
#vout		Number of transaction inputs in vout.
	nValue	Amount of 10^{-8} BTC.
	scriptPubkeyLen	Length of scriptPubkey field in bytes.
	scriptPubkey	Script satisfying conditions under which the transaction output can be claimed.
nLockTime		Timestamp past which transaction can be replaced before inclusion in block.

While spending user's bitcoins, the basic logic like that: Incoming transactions to user are references to spend them. But the same incoming input a.k.a bitcoins cannot be spent more than one times. This is controlled by miner while verifying transactions. User spends his bitcoins with referencing incoming transactions as inputs for his transaction. In transaction structure, vin[] array shows the incoming transactions to him for this transaction, vout[] is the array of addresses and values for sending bitcoins. Figure 2.4 shows how to a coinbase transaction is used in a regular transaction.

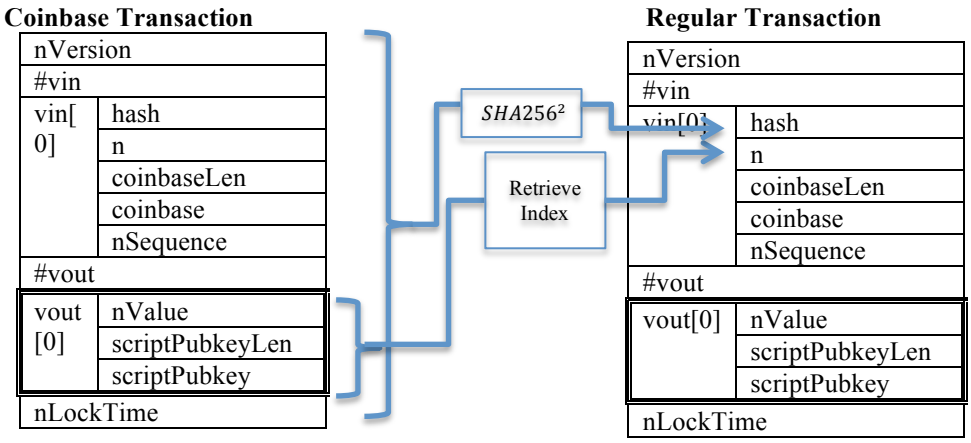

Figure 2.4 Coinbase transaction to regular transaction [15].

2.6 Transaction Verification

During transaction verification, it is checked whether input transactions are unspent, signature is matched with the given public key or given public key hash is matched with the public key and signature is matched to this public key. To check these signature and key matching, Bitcoin uses Forth-like scripting language called Script.

- **Script**

Script is a stack-based Turing-incomplete programming language designed for Bitcoin to check validity of transactions. Script is used to encode two parts: Challenge script a.k.a *scriptPubkey* is used to specify conditions so that transaction can be verified. Response script is created by receiver to verify validity transaction. All script is run left to right.

Example of pay-to-pubkey transaction verification:

If the signature given in *scriptSig* is signed with private key corresponds given in *scripPubKey* public key, it returns true. `scriptSig` is provided by receiver. `<pubKey>` is receiver's public key and `scriptPubKey` is provided by sender. `<pubKey>` is receiver's public key. Table 2.3 shows an example of script execution process.

```
scriptPubKey: <pubKey> OP_CHECKSIG    (Challenge Script)
scriptSig: <sig>                      (Response Script)
```

Table 2.3 Example script execution process [15].

Stack	**Script**	**Explanation**
Empty	<sig><pubKey>OP_CHECKSIG	Merge two parts.
<sig><pubKey>	OP_CHECKSIG	Put variables into stack.
TRUE	Empty	Verify.

This control is to understand if address is owned by receiver or not. If address is proved owned by receiver by validating receiver's signature, it means that transaction is addressed to right receiver address.

2.7 Coin Mining

Mining means that miner finds a valid block for decided difficulty which contains broadcasted transactions by users. Mining is the unique way of minting in Bitcoin system. While miners validate transactions, they earn transaction fees as kind of incentive for miners. After collecting broadcasted transactions, Script verifies transaction like in transaction verification process. Then, input transactions are controlled if they are unspent before. Taking last mined block, miners creates the block data and starts to find target hash with different nonce values starting from 0. When miner finds the target hash means solving Poof-Of-Work, it broadcasts the block to the network and others verify the block with found nonce and they accept the new block and every full node add the new block to their local blockchain.

Mining is a kind of race condition. Because every miner or mining pool tries to find a valid block to get reward and transaction fees by paid owner of transactions. While during this race, blockchain has forks. It means more than one valid block can be found at the same time. Which one is chosen as main chain will be decided in next block. The main idea: the longest chain is the main chain and miner continues to mine from that block as in figure 2.5. The shorter block remains *orphaned blocks* as in figure 2.2.

Mining means transaction validation at the same time. Because while miners are trying to find a next block, they include all transactions published at that time. Then, miners check if these transactions are valid or not. It means that transaction owner has enough funds to send to another user. After that, miner takes these valid transactions and constructs a Merkle tree and adds this Merkle tree to transaction data and tries to find a block with different nonce values at a definite difficulty declared by Bitcoin system. If miner finds a block including these transactions, it means that these transactions get one confirmation. Then, miners try to find next block, which includes hash of this block, and it continues in this way. It is considered

that a transaction can be accepted as spendable input for security of users, it needs to get at least 6 confirmations. It means that this transaction must be included by a block and after this block, 5 more blocks must be found tied to this first block in chained manner like in figure 2.5. Six confirmations are advised to users by Bitcoin system to guarantee user funds.

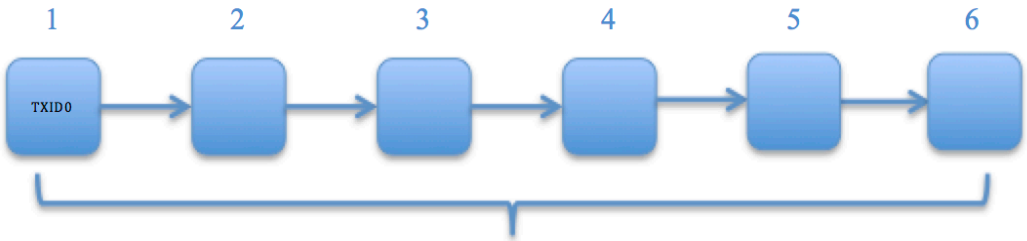

Figure 2.5 Txid0 gets 6 confirmations (it takes approx. 60 minutes).

Mining started with CPU (Central Processing Unit). But with time it is realized that CPU mining is not efficient and financially loss. Because CPU is not good at running hash algorithms. Then, miners moved to GPU (Graphical Processing Unit) which has a suitable architecture for hashing algorithm. But with time, difficulty got higher and GPU became inefficient for mining. After GPU, FPGA (Field Programmable Gate Array) took action on mining. FPGA is faster than GPU in mining process and consumes small amount of power. But, FPGA was not enough mining. So, ASIC (Application Specific Integrated Circuit) mining is started. A special type of ASIC miner is developed for hashing algorithms in Bitcoin system. ASIC miner has major speed in mining rather than other mining types. A single miner started to not create enough hashrate for mining. Because a huge hash rate was need to run. So, miners get together and created *mining pools*.

2.7.1 Hash Function in Bitcoin : SHA2-256

In cryptocurrency, hash function is used for mining for blocks. When the target hash value with the desired difficulty is found by miner, miner gets the reward as bitcoin supplied by Bitcoin system. Cryptographic hash functions are used for proof-of-work is called hashcash. SHA-256 is utilized as cryptographic hash function in Bitcoin.

SHA-2 (Secure Hash Algorithm 2) is developed by NSA in 2001 [63]. SHA-2 is an improved version of SHA-1. SHA2-256 has 256 bit hash value. SHA-2 family has SHA-224, SHA-384, SHA-512, SHA-512/224, SHA-512/256 versions.

1. SHA-256 provides an estimated preimage resistance of 256 bits [64].
2. SHA-256 provides an estimated second preimage resistance of 256 bits [64].
3. SHA-256 provides an estimated collision resistance of 128 bits [64].

To the best of our knowledge, any collusion has not been announced for SHA256 yet.

In Bitcoin, SHA256 is used double times. It is shown with $SHA256^2$. It is used to find the target hash of block header determined by the system automatically. This process called coin mining process. If miner can find a block satisfies the given difficulty, publishes in the network and network verify it is a valid block, miner gets reward because of creating a new block in blockchain.

$$SHA256^2 (nVersion \,||\, HashPrevBlock \,||\, HashMerkleRoot \,||\, nTime \,||\, nBits \,||\, nNonce) < Target\ hash \quad (2.1)$$

For 396826 block [65] which is the last mined block at the of writing [5] as an example of a mined block:

In Python using hashlib library [66],

1. Hex of all parts are reversed with 2 characters.
2. Concatenate all parts as in below order.
3. 3.Decode the concatenated value from hexadecimal to ASCII.
4. 4.Take double SHA256 hash the last value.
5. Result is the reverse of the desired target value. Difficulty is the number of 0 bits at the end of the result value.

```
header_hex = ("04000000" +
"a902c3f3a26134acc6fffd3cf117f714f4253b945388ae00000000000000
```

[5] Date of writing :05.02.2016

```
0000" +
"138defc1c2b9b6ca1375f2a8a55afa0c01bc0e59d05d36676efc05dc49ac
c8f5" +"A77FB456" +"f0280918" + "D2C74C56")

header_bin = header_hex.decode('hex')

hash =
hashlib.sha256(hashlib.sha256(header_bin).digest()).digest()
```

Hash value of this block header is found as below;8e69a89833468273760041e084186a521ad62d1015715b080000000000000000

After reversing it with 2-character holding:
000000000000000085b7115102dd61a526a1884e04100767382463398a8698e

This block is mined as explained above using a nonce value starting from 0 to unknown number till the target value is found. This value is smaller than given difficulty by the system. It starts with 17 zeros. If hardness of mining gets higher that means target gets smaller, number of zeros increases.

Also $SHA256^2$ is used to compute transaction id of a transaction like $TxId = SHA256^2(Transaction)$ which is located in Merkle tree.

2.7.2 Mining Pools

In mining, difficulty is getting higher day by day. So, a unique miner cannot find a valid block by itself. It is called *solo mining*. Because difficulty is so high that a normal devices also ASIC miners cannot find the target hash. So, miners collaborate in mining pools [67]. They share their hash rate power and reward. There are many types of pools according to their sharing system. Mining pools were not foreseen in Bitcoin protocol.

- **Miner's Pool Choice**

Miners can join mining pools and they contribute to pool hash power. There two type of pools called open pools and closed pools. Open pools have public web interface for miner to log in to register. Miners registers via this interface and they

are assigned a mining tasks. Miners find proof-of-work, submit these proofs to pool manager and take their shares. Closed pool members trust each other so smaller than open pools.

Miners face a dilemma that they prefer attack an open pool or not attack. Eyal called this dilemma "miner's dilemma" similar to prisoner's dilemma [68]. Sometimes miners try to attack other open pools. It means that they join that pool and they do not contribute the pool but they do not seem like that. Eyal showed that an open pool can attack another open pool and have more profits. But, if both of open pools attack each other, both can have profits less than if none of them attacks [68].

It was believed that miners do not attack other open pool so that their profit does not decrease. It was considered an unprofitable method. But, Eyal showed that if there is an agreement to not attack each other, they earn profit with only this way. Because, if a pool attacks its peers, peers start to attack others and this ruin pool profit. So, miners prefers closed pools so that pool cannot be attacked by outside malicious miners. But closed pools are smaller than open pools because of trust issue. Eyal shows that a pool earns more profit when it attacks to an honest pool all in pool size conditions.

Pools can be too large that they come to 51% limit. Even smaller pool size can be dangerous for Bitcoin network. Selfish mining shows how it can result serious consequences. So, miner distribution over pools is very important.

Block withholding attack is a kind of successful attack in the pool system. Miner holds its block and reveals when it has longer block fork. This will result double spending.

2.7.3 Mining Pools Types

- **The slush approach:** It depends on score-based method. Older shares have lower scores. Recent shares have higher weight on score.
- **The Pay-per-Share approach:** PPS pays to miner an instant amount for each share solved by miner.

- **P2Pool approach:** In P2Pool, miners work on a share-blockchain similar to Bitcoin blockchain. When a block is found, reward is shared among the most recent shares [69].
- **Luke-Jr's approach (Eligius):** All miners are paid via the Generation transaction who have the minimum payout [70].

Other mining pool ideas: **CPPSRB** - Capped Pay Per Share with Recent Backpay, **DGM** - Double Geometric Method, **POT** - Pay On Target, **PPLNS** - Pay Per Last N Shares. **PPLNSG** - Pay Per Last N Groups, **Prop.** - Proportional, **RSMPPS** - Recent Shared Maximum Pay Per Share, **Score** - Score based system, **SMPPS** - Shared Maximum Pay Per Share.

2.7.4 Proof-Of-Work

Bitcoin mining depends on finding a hash with calculation (2.1) below target hash. In every attempt with a nonce, miner calculates (2.1) and reverses the byte order of result and checks if the final hash lies below target hash. Target hash is specified in blocks called *nBits* field.

$\mathrm{SHA256}^2$ (nVersion || HashPrevBlock || HashMerkleRoot || nTime || nBits || nNonce) < Target hash **(2.1)**

2.7.5 Difficulty / Target / nBits Field

The *nBits* field in block header stores a compact representation of a target value T as $0xh_0h_1h_2h_3h_4h_5h_6h_7$ where h_i represents a hexadecimal digit. Target value is 256-bit number. But in nBits field, compact form of target value is used as encoded with only 8 hexadecimal digits. Computing long form of target value from its compact form is like below. Long form is used when the new difficulty is calculated.

$$0\mathrm{x}h_2h_3h_4h_5h_6h_7 \cdot 2^{8\cdot(0xh_0h_1-3)} \quad \textbf{(2.2)}$$

Upper bound of target value is 0x1D00FFFF and there is no lower bound. Genesis block is mined with the maximum target that means minimum difficulty. The lower target value, the difficulty gets higher. Because, if target value gets smaller, number

of zeros at the beginning of target value increases. So miner must find a target hash starts with more number of zeros. For example, if target value goes to 0, it means that miner must find a hash contains nearly 256 bit zeros. That means difficulty is very high. Because finding a hash value starts with 256 bit zeros is really hard work.

Difficulty is a measure of how difficult it is to find a hash below a given target [71]. In every 2016 blocks (2 weeks for 10 minute mining time), difficulty is readjusted according to block mining time to fix it to 10 minutes. Next target T' is computed using present target T like below. t_{sum} is difference of timestamps of block headers.

$$T' = \frac{t_{sum}}{14 \cdot 24 \cdot 60 \cdot 60\ s} \cdot T \qquad \textbf{(2.3)}$$

At the time of writing[6], *nBits* is 403153172, 1807A114 in hexadecimal. Current difficulty is 144116447847.34866 [72].

2.8 Wallet

Wallet stores public, private key and address of user in hardware of user and creates transactions to send and take transactions from others. Mainly, there are two methods for wallet. These are hot wallet and cold wallet. Hot wallet is connected to the Internet. It is open to online attacks. Cold wallet is offline wallet which does not have any connection to the internet. So, online attacks are not useful to steal user private key. For example; USB hard drive, other data storage devices, paper wallet, air-gapped system wallet.

Software wallets, website wallets and paper wallets are different usages of wallet which are used for different security reasons. Wallet secures private key of user against theft. If private key is stolen, coins of the owner of private key can be used by others. Some popular Bitcoin wallets are Bitcoin Core, Multibit, Electrum, Hive, Armory, Blockchain Wallet, Coinbase, Coinkite, BitAddress Paper Wallet, Pi-Wallet.

---

[6] Date: 11.02.2016 21.44(GMT+2.00)

2.9 Peer-to-Peer Network

Peer-to-Peer network is a distributed resource sharing architecture which is held by more than one computer system without any central node over the internet or directly connected each other via special network protocols. Every peer can contribute the network or make use of system resources. Every connected node is a client and server at the same time. In virtual currencies, peer-to-peer network is used to set up a decentralized system that is nature of VCs. Another application of P2P networks is BitTorrent [73] which is one the most popular file sharing protocol with transferring large files over the internet.

Similarly BitTorrent, Bitcoin is a P2P payment network, which shares transactions over the Internet on a public ledger called blockchain secured by cryptographic protocol. Bitcoin network runs over TCP. Every node has 117 incoming TCP connections and 8 outgoing TCP connections by default. It has a random topology that nodes join the network whenever they want. For leaving network, there is not a way to leave network explicitly. If a node is not heard for 90 minutes, other nodes forget that node [74]. In Bitcoin network, all nodes are equal. There is not any hierarchy, and there are not any special nodes or master nodes in it. Users send bitcoins other peers with sharing transaction with all Bitcoin network. Thanks to this, all nodes are informed about users' bitcoin balance with checking spent and unspent transactions of users. This also prevents *double spending* that is user's attempt to spend same money for different services, goods etc. If network realize that the bitcoins of user are spent before (because every transaction is published on the network), the second attempt is failed and not validated by miners who verify transactions.

There are full nodes (full node clients) and lightweight nodes Simplified Payment Verification (SPV) Clients. Full nodes download every block and transaction from the network. But, lightweight nodes do not store all blocks. Most of the network consists of lightweight nodes because of big amount of blockchain size. Now,

blockchain size is nearly 53. 78 GB [7]. Every full node must store this amount of data to validate transactions and coins. SPV clients download a complete copy of the headers for all blocks in the entire block chain not the entire blockchain. SPV client determines the validity of a transaction by checking how many blocks have been mined on top of the block where the transaction is included with the name *block depth validity check*. SPV clients must download nearly 30 MB data nowadays. At the time of writing, there are 5835 reachable full nodes in Bitcoin network[8]. Also, in Bitcoin network, is allowed 7 transactions per second because of 1 MB limit of block size by Bitcoin protocol [75]. But, in the future it is supposed that the number of transactions which need verification in Bitcoin network will increase and in order to handle this issue, block size will need to be increased. But this change will result scalability problem because of increasing blockchain size. Scalability problem is explained in third section.

In Bitcoin network, for inbound connection, peers listen to 8333 port. When peers establish a connection, they have an application layer handshake with version and *verack* messages. Peers transmit a heartbeat message to keep the connection alive after they exchange messages with neighbors. Every peer broadcasts its IP address in *addr* message in every 24 hours in the network. So, peers keep data of not directly connected active peers including their IP addresses and a timestamp. Peers use three ways to find neighbors during exchanging *addr* messages: DNS, IRC, asking neighbors. Bitcoin version 0.6 uses DNS as default [76]. Donet et al. discovered 872, 648 IP addresses in total after 37 rounds in 37 days [77]. Because of dynamic IP addressing of some peers, peers in the network are not stable. https://bitnodes.21.co presents data about Bitcoin nodes.

---

[7] For more information: https://blockchain.info/charts/blocks-size at 08.02.2016 14:33:01 GMT+0200 (EET).

[8] For more information: https://bitnodes.21.co/ Reachable nodes as of 08.02.2016 14:33:01 GMT+0200 (EET).

2.10 MainNet

Mainnet is the Bitcoin's main network where real Bitcoin transactions run. Bitcoins have real economic value in this environment. Real transactions are saved in blockchain. This blockchain has all real world Bitcoin transactions from the scratch. For developers, testing their applications is very expensive because of need of real bitcoins. Developers need to have real transactions with real bitcoins. So, for developers testnet is a cheaper and faster way to simulate applications.

- **Testnet**

Testnet is a kind of Bitcoin environment for developers to try Bitcoin network where the bitcoins cannot be spent and have value like in Bitcoin mainnet.

- **Regtest Mode**

Bitcoin Regression test mode gives opportunity to have experience to create new blockchain with the same as testnet. Many developers prefers regtest mode to develop new applications for Bitcoin [78].

2.11 Bitcoin Client

Bitcoin Client is software for end user to generate private key, public key and addresses. It is used for transactions between users and keeps useful information about Bitcoin network. It must have wallet security to secure private key of user and implementing Bitcoin network protocols safely.

Satoshi client or satoshi code is the original Bitcoin client. Bitcoind is the second Bitcoin client which does not have GUI (Graphical User Interface) with JSON-RPC interface. Bitcoin Core is the third bitcoin client which is C++/Qt based tabbed with GUI. Bitcoin-QT has been renamed as Bitcoin Core since version 0.9.0. Multibit, Electrum, Armory, Bitcoiner, btcd are another mostly used Bitcoin clients.

2.12 Hard Fork and Soft Fork

Hard fork and soft fork are alterations of the Bitcoin protocol. Types of changing software must be considered carefully according to way of alteration. While some researchers claim that soft fork is safer than hard forks [79], others accept hard fork is better than soft fork [80].

- **Hard fork**

Hard fork is a kind of alteration of Bitcoin protocol with removing rules from Bitcoin protocol. Hard fork makes previously invalid blocks or transactions valid. Because deleting some rules removes some obligations. So, some invalid blocks/transactions can be valid anymore. This kind of alteration can be about block structure, difficulty rules. Hard fork increases number of valid blocks/transactions in Bitcoin network. Also, hard fork is not *backward compatible.* Backward compatible means that after alteration of rules, old nodes (do not know about rule change) do not accept new blocks as valid. Because there is an invalidity for old nodes in the created new block. Lets say; in a block, before rule removing, there must be A, B, C rules. Old node checks if the block is suitable for those rules. After rule alteration, C rule is deleted and old nodes do not about them. But old node still checks if a block is suitable for A, B, C rules. But after alteration, new node has just A, B rule, not valid for C rule. So, old node decided new node as invalid because of node does not have C. So, hard fork is not backward compatible.

- **Soft fork**

Soft fork is a kind of rule alteration of Bitcoin protocol with adding new rules to Bitcoin protocol. Soft fork makes previously valid blocks or transactions invalid. Soft fork is about adding new rules like new transaction rules. Pay-to-Script-Hash (P2SH) was a soft fork in the network with BIP16 [180]. It allowed transactions to be sent to a script hash instead of a public key hash. Soft fork increases invalid blocks. Also, soft fork is backward compatible. It means that old nodes (do not know about rule adding) still accepts new blocks created according to new rule alteration. Lets say; before alteration, blocks/transactions must be suitable A, B rules. After adding

D rule to Bitcoin network, old nodes check just A, B rules if the block/transaction is valid according to A, B rules. So, old node sees that new block/transaction is valid for A, B rule. So, soft fork is backward compatible. On the other hand, Mike Hearn claims that backward compatibility term is used as a wrong term for this scope. He offers *forwards compatibility* [80]. He thinks forward compatibility means that old software continues to accept block/transaction produced by new software.

2.13 Who is Who in Bitcoin?

- **Developers of the Core Bitcoin Client**

Wladimir van der Laan is the maintainer of Bitcoin Core. Key contributors are Gavin Andresen, Matt Corallo, Corey Fields, Jeff Garzik, Luke-Jr, Gregory Maxwell, Peter Todd, Pieter Wuille. These developers contribute to Bitcoin core development. They check proposals for Bitcoin.

- **Large Miners and Mining Pool Operators**

AntPool, BitFury, BTCChina, F2Pool, KnCMiner, Slush, BW Mining are some large miners in Bitcoin. Miners verify transactions and broadcast transactions and blocks. They vote proposals using block by writing their preferences in it. If a proposal gets majority votes, then proposal can be added to Bitcoin.

- **Users and Wallet Providers**

Armory, Bitcoin-QT, BitGo, Blockchain, Electrum, MultiBit HD, Bither, mSigna, Coinomi, GreenBits are some of the wallet providers of Bitcoin. Users use wallets to relay requests, transactions to Bitcoin network.

- **Exchanges**

Bitstamp, BTCChina, Coinbase, Coinsetter, Cryptsy, Kraken are some exchange sites allows users trades between Bitcoin and other currencies.

CHAPTER 3

ALTERNATIVE COINS

3.1 Zcash

Zcash is the new name of Zerocoin and Zerocash project [5, 6]. Zcash is a full-fledged digital currency which has implementation of the "zerocash" protocol published as alpha version for community to develop and watch its process. Zcash team[9] aims at creating private and anonymous cryptocurrency system. “All coins are created equal.” is motto of Zcash[10]. Zcash was launched as testnet in 20 January 2016 by Zcash Electronic Coin Company leaded by Zooko Wilcox-O'hearn. They published all code on GitHub[11] for public interest. Also, blockchain for Zcash is launched by the company to test Zcash transactions and simulation before opening to public for real usage[12]. Users can mine “testnet-bux” to experience Zcash environment. Currency symbol of Zcash is ZEC.

Zcash started with Zerocoin project to fix privacy problem which is a major weakness in Bitcoin. Zerocoin was proposed as an extension to Bitcoin to provide untraceability to user transactions on the blockchain.

Zerocoin allows user to mix own coins to provide privacy. After Zerocoin was developed, new project called Zerocash is developed by a new team. Zerocash allows using direct private payments among users with private transaction value. Zerocash hides origin, destination and amount of transaction, while Zerocoin keep secret only origin of transaction.

---

[9] Zooko Wilcox,Eli Ben-Sasson,Alessandro Chiesa,Christina Garman,Matthew Green,Ian Miers,Eran Tromer,Madars Virza,Nathan Wilcox,Daira Hopwood,Sean Bowe,Taylor Hornby,Jack Grigg, Gavin Andresen,Vitalik Buterin,Andrew Miller,Arthur Breitman,Joseph Bonneau

[10] Zcash website https://z.cash/ Retrieved 27 February 2016

[11] https://github.com/zcash/zcash

[12] For more information: http://coin.cell.systems/

3.1.1 Zerocoin

Zerocoin is a kind of decentralized mix and uses digital commitments, one-way accumulators and zero-knowledge proofs and has some similarity between Auditable, Anonymous Electronic Cash system proposed by Sander and Ta-Shma [81]. Zero-knowledge proofs which are based on Schnorr technique and non-interactive zero-knowledge proofs are used in Zerocoin [82, 83]. Also, Zerocoin does not use a new trusted third party to provide security. Zerocoin is easy to integrate into Bitcoin as an extension. Basic idea of Zerocoin is like below:

To mint a Zerocoin:

1. Alice generates a random serial number *S* and secret *r*.
2. Alice commits *S* with using r. Commit (*S,r*) = *C C* corrresponds a zerocoin.
3. Amount of *C* is added to the Zerocoin escrow pool.

To redeem a Zerocoin into Bitcoin (preferably a new Bitcoin address):

1. Alice must prove 2 things using zero-knowledge proofs.
   a) Alice proves *C* that belongs to set of $(C_1, C_2, \ldots\ldots, C_n)$ by keeping secret which one is *C*. This is done by one-way accumulators.
   b) Alice proves that she knows a number *r*, that matches the serial number *S* corresponds to a zerocoin.
2. The proof and serial number *S* are posted as a zerocoin spend transaction, and miners verify the proof and that the serial number *S* was not spent in a previous transaction.
3. After verification, the transaction is posted to the blockchain, and the amount of bitcoin, which is equal to the zerocoin value, is transferred from the zerocoin escrow pool.

At the end of this process, C and S cannot be linked to each other and this property provides privacy to Zerocoin.

While Zerocoin provides privacy, Zerocoin has disadvantages in terms of big amount of proof size 40KB and slowness in proof process like taking 2 seconds to verify. So,

they improved Zerocoin and the new version was dubbed Zerocash. Zerocoin was firstly used in an altcoin called as Moneta in 18 December 2015 [84]. While figure 3.1 shows Bitcoin money flow in blockchain, figure 3.2 shows transaction flow in Zerocoin.

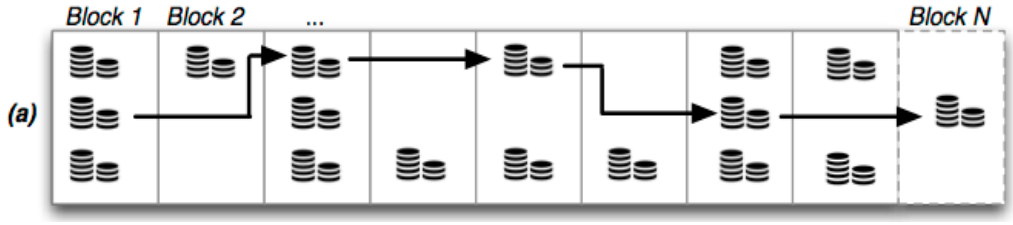

Figure 3.1 Transaction flow in Bitcoin blockchain [6].

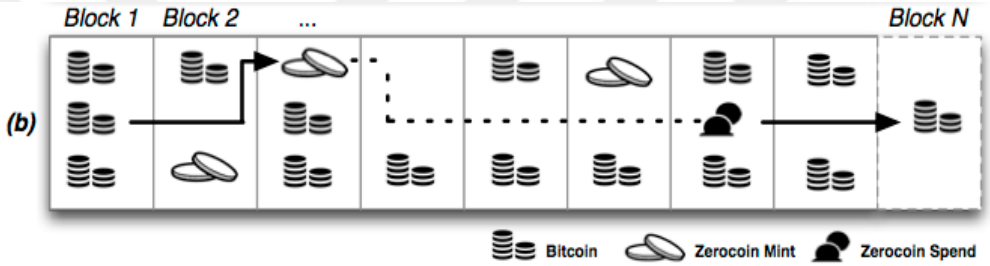

Figure 3.2 Transaction flow in Zerocoin [6].

3.1.2 Zerocash

Zerocash is rooted from zero-knowledge Succinct Non-interactive ARguments of Knowledge (zk-SNARKs) and has smaller data sizes with the percentage of nearly 98% in contrast to Zerocoin [5]. Zerocash is an improved version of Zerocoin with origin, destination and amount of transaction privacy. There are two types of coin in Zerocash. The first one is zerocoin the anonymous one, the second one is the non-anonymous one called basecoins. Basecoins can be converted to zerocoins and vice versa. Zerocoins can be merged and split into pieces without revealing information about amount of them. Zerocash is a type of decentralized anonymous payment scheme (DAP scheme) which is used for private transactions. There are two types of transactions in Zerocash called *mint* and *pour transactions*.

- **Mint Transaction:** Converting a non-anonymous bitcoins from a bitcoin address to at the same amount of zerocoins owned by a Zerocash address is called as mint transaction. Mint transaction consists of cryptographic commitment scheme which includes coin's value, owner address, and (unique) serial number to convert a new coin by keeping secret this information. SHA-256 hash function is utilized in this commitment scheme.
- **Pour Transaction:** Pour transaction is used to make private transactions with coins. Pour transaction consumes input coins by just revealing their serial number not other information. It includes zero-knowledge that proves;
    1. User owns (up to) 2 inputs,
    2. Each input coin appeared in previous mint transactions or pour transactions (as output)
    3. Equality between amount of inputs and amount of outputs
- **Verifying Zerocash Transactions:** Anyone can verify mint transactions because of containing commitment in itself. Anyone verifies that committed coin has the claimed value. Also, anyone can verify pour transactions because of containing zero-knowledge in itself. Anyone can verify pour transaction that claimed proof is valid. In this step, Zerocash utilizes zk-SNARKs. Because zk-SNARKs are easy to verify and fast like in milliseconds and has smaller proof sizes like 300 bytes [85].

3.2 Ripple (XRP)

Ripple is a distributed, open source network protocol for payments which exchanges between different currencies. Ripple network acts as a decentralized currency exchange. It is accepted by CGAP in 2015 that Ripple payment protocol is payment protocol of the Internet like SMTP (Simple Mail Transfer Protocol) of email protocols [86]. Its native cryptocurrency called Ripple or XRP is pre-mined digital currency. It is the cryptocurrency that has the third-largest market capitalization[13].

Ripple network relies on public ledger that stores information about all Ripple accounts. Ripple network is managed by independent peers including banks, market

[13] For more information: https://coinmarketcap.com/

makers. There are 100 billions of XRP in the system and XRP only exists in Ripple network. Transactions are validated by the consensus. If "supermajority" is supplied for a transaction, this transaction becomes validated. Transactions are not applied to Ripple ledger as soon as they are published. After validation, they are stored in the ledger by checking only validated ledgers. Ripple ledger stores ledger number, account settings, trust lines balances, transactions and timestamp. Ledgers are linked each other in date order with cryptographic ties. Each transaction is signed by its owner and the unique way to change account values. Figure 3.3 shows Ripple ledger structure

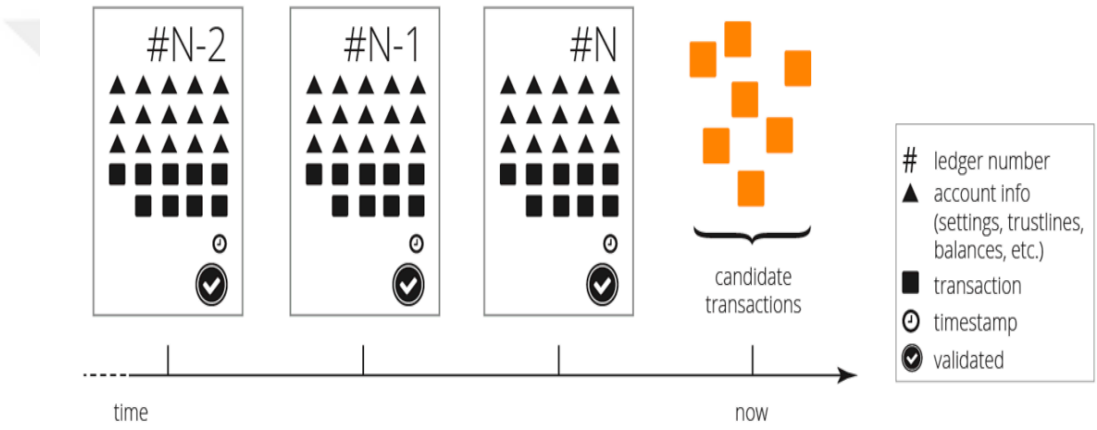

Figure 3.3 Ripple ledger structure [87].

Transactions are verified by consensus not by mining. Transactions are irreversible and can be verified in seconds. Nodes do not need to download all blockchain. Thus, they can be ready in seconds. Users must trust gateways. Because users deposit their funds to gateways and gateways do necessary exchanges for them. Thus, user must put a limit to this trusts for each currency. Also, user can allow more than one gateways and apply "rippling" on these gateways by allowing funds to switch between gateways. But user's total balance remains same.

It allows transferring money with negligible fees and wait-time between users. Transaction fee is very small amount with XRP and this amount is destroyed later. This precaution is for attackers who want to flood system with too many transactions

to cause a network deadlock. Figure 3.4 shows connection of sending and receiving bank via Ripple gateways.

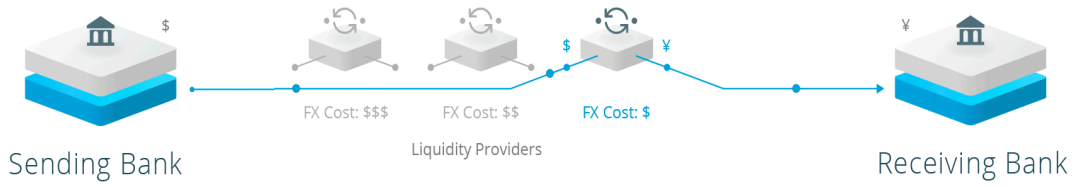

Figure 3.4 Ripple network flow [ripple.com].

In Figure 3.4, sending party sends its funds as USD, receiving party takes it as Yen. In the middle of sending and receiver there is an exchange gateway to convert USD to Yen by using XRP. XRP is a bridge currency in Ripple network. Gateways are businesses to exchange money from a currency type to another. Bitstamp, Gatehub, Bluzelle are some of the popular gateways for Ripple network.

3.2.1 Ripple Protocol Components

Ripple networks consists of different components.

Server: Servers run Ripple server software to conduct consensus process like validation. But clients run Ripple client software only for sending and receiving funds in the network.

- **Ledger:** Ledgers stores user's account balance and is updated by nodes when new transactions are validated by using consensus.
- **Last-Closed Ledger:** Last-closed ledger is the final state of the ledger. After last consensus process run, the ledger is updated and called last-closed ledger.
- **Open-Ledger:** Open-ledger is the open to coming transaction ledger owned by each node. Before transactions are validated by using consensus, this ledger is called open-ledger. After consensus is conducted on this ledger, it becomes last-closed ledger.
- **Unique Node Lists:** Every node, server in the network, stores own node list which contains all other servers conducting consensus protocol. Every node

determines own trusted node list not to let other nodes create a malicious majority for fraud to the network.

- **Proposer:** Every node broadcasts transactions and a node considers some transactions broadcasted by nodes in its UNL. Also, Unique node lists are called UNLs.

3.2.2 Ripple Transaction Protocol (RTXP)

Like SMTP creates a mailing ecosystem, RTXP creates a transaction network between users. RTXP is a financial protocol that serves communication inter-financial systems.

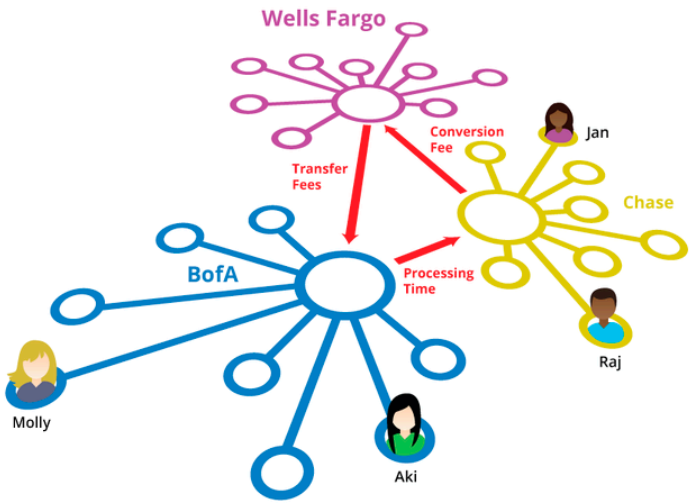

Figure 3.5 Ripple payment network [88].

RTXP runs on Ripple network. Ripple network consists of nodes, which validates transactions. Client applications create transactions and send transactions to nodes to be validated. Some transactions are validated by consensus in candidate transactions and created a new ledger containing these validated transactions. Ripple network adjusts transaction fees dynamically. Ripple consensus is detailed in the whitepaper of Ripple Protocol Consensus Algorithm [8].

3.2.3 Ripple Protocol Consensus Algorithm

RPCA is run in every few seconds by nodes to maintain network consistency about transaction validations. If a consensus is decided by majority, ledger is closed and

new ledger consensus is started by nodes. If consensus algorithm is run successfully and there is not any fork in the network, all nodes store the last closed ledger. RPCA runs below [8, 89]:

1. Each server takes all validated transactions to prevent revalidating old ones. If there are new transactions published by server's end nodes, server publish these transactions to the network as candidate set.
2. Each server merges all candidate set published by servers on its UNL. Server votes for all transactions if they are reliable or not means that valid or not.
3. Transactions get enough "yes" vote continue in the algorithm. If some of them cannot enough "yes" vote, they wait for next open-ledger.
4. If a transaction gets confirmation greater than 80% of a server's UNL, this transaction is added to confirmed list. After checking all transactions, ledger is closed and called last-closed ledger.

3.3 Peercoin

Peercoin is peer-to-peer decentralized crytocurrency that uses hybrid system that uses proof-of-stake (PoS) and proof-of-work (PoW) [7]. Proof-of-work is widely known technique used by crytocurrency like Bitcoin for mining and to prevent denial of service attacks or spam to system. Proof-of-stake is alternative to proof-of work in terms of aim. But PoS uses coin age idea not similar to hashcash PoW of Bitcoin.

3.3.1 Proof-of-Stake

PoS aims to create a consensus and prevent double spending. It was proposed firstly in BitcoinTalk by a member with username "QuantumMechanic" [90]. Main idea is to accumulate the greatest coin age in the network and create a block by proving ownership of these coins. If nodes in network verify and accept this new block, owner of coin age gets reward like in Bitcoin system and miner's coin age starts from zero. But difference between PoS and hashcash PoW, hashcash has high energy and time consumption. Because in PoS they calculate only one hash, in hashcash, miner tries to find a valid block that is suitable for a certain difficulty. So, hashcash miner spends large amount of energy till the time finds a valid block. PoS secures network

and allows controlled block generation and used by Blackcoin, Nextcoin, Bitshares and Qora as first implementations of PoS.

3.3.2 Coin Age

Coin age idea relies on amount of user's holding coins. Firstly, Satoshi Nakamoto defined coin age in 2010. Basic idea behind coin age is duration of coins stayed in address multiplication with amount of coins. For instance; If Alice takes 10 coins from Bob and holds them in her address during 30 days, coin age of Alice's coins is 30x10=300 coin-days. If Alice spends these coins, it means that she destroyed or consumed her coin age. To compute coin age, there is timestamp data in blocks and transactions. Also, there is another feature of PoS in Peercoin that users are rewarded 1% annual interest for storing their coins on their stack. This is a kind of incentive to stay in the network. Peercoin does not use transaction fees (0.01 PPC/kB defined in the Peercoin protocol) as fund, it destroys transaction fees before reaching miner. This regulates transaction volume and hindering spam. Also, Peercoin does not have a hard limit in the number of coins will be created in the future. So, destroying transaction fee is a kind of deflationary effect. Also, this system aims to long-term scalability.

3.4 Dash

Dash, previously known as Darkcoin is open source decentralized peer-to-peer crytocurrency based on CoinJoin launched 18 January 2014 [42]. Darkcoin, currently Dash is the first attempt to solve privacy problem in Bitcoin by rooting from Bitcoin. Dash is the fifth cryptocurrency that has the most market capitalization [14]. Dash has an implementation of CoinJoin called *Darksend* to create private transactions. Dash uses X11 hashing algorithm instead of SHA-256. User have private transactions which others cannot find a link between owner or owner's address and owner's transaction. In Darksend, blind signatures are utilized to prove transaction is owned by a particular user so that any master node or other nodes cannot know which output is owned by which input.

---

[14] For more information: https://coinmarketcap.com/

3.4.1 Darksend

Darksend is extended version of CoinJoin mixing service explained in the forth chapter under privacy title. Darksend adds new improvements like decentralization, strong anonymity. CoinJoin allows users have more than one operation in one transaction. Two users send their funds to two different addresses in one transaction so that it is not found who sends to whom. Thanks to CoinJoin, any third part cannot trace any user's transaction history clearly. But under some conditions, it is possible. It is explained under privacy title in the forth chapter.

Darksend aims privacy of user. So, it uses chaining approach that funds are sent to sessions one after another. Each session contains three clients. So, with increasing chain depth, traceability of any transaction decreases. Table 3.1 represents chain depth and probability relation.

Table 3.1 Chain depth and probability.

Chain Depth	**Probability to Trace a Transaction**
2	1/9
3	1/27
4	1/81

3.4.2 Masternode Network

Masternodes perform Darksend transactions on a decentralized network by signing transactions. Masternodes process more than one round mixing for anonymity. Masternodes are chosen randomly to make coin mixing. Masternodes are untrustworthy parties because they can steal user funds. Mixing services are not reliable because of this reason. Masternode network must provide a high synchronization and fast propagation of data in the network. For preventing sybil attacks by malicious Masternode candidates, they must have 1000 Dash to be a masternode [189]. Masternodes need incentive to perform coin mixing. Their incentive is to take 45% reward of mining.

Mining difficulty is adjusted by Dark Gravity Wave algorithm. Mining reward is adjusted by Moore's law $2222222/((\text{Difficulty}+2600)/9)^2$ [91].

For double spending problem InstantX is developed. It is a service that locks inputs into specific transactions. After verification of consensus, it rejects conflicting transactions. It provides faster confirmation. There are 3580 Masternodes in the network at the time of writing [92]. If a malicious community wants to take control of 50% of the masternode network, it needs to buy high amount of Dash coins like 3,580,000 to be accepted as a masternode. This will raise Dash price and it will be harder to buy. So, it is hard to take control by a malicious third party.

3.4.3 Mining

Dash uses CPU and GPU mining. Mining steps are similar to Bitcoin by using PoW idea. But Dash uses X11 hashing algorithm. X11 uses 11 different hashing algorithm rounds like blake, bmw, groestl, jh, keccak, skein, luffa, cubehash, shavite, simd, echo. Thus, it has higher complexity than other hashing algorithms. In case of breaking one of the hashing algorithms, X11 is still secure. Thus, it handles single point of failure risk. Also, X11 is ASIC resistant. ASIC miner is believed that it centralize hashing power. So, for preventing centralization of hashing power, there is not any implementation of X11 for ASICs. There are two ways to mine a Dash coin called solo mining and mining pools mining. Solo mining is least used method because of less hash power of individuals. Mining pools are mostly preferred method to find a block faster by merging their hash power. After they find a block, they share mining reward among them. Peer-to-peer pools are more reliable in mining pools in terms of security of funds. Centralized pools are more likely vulnerable to 51% attacks.

CHAPTER 4

COMPARISON OF BITCOIN AND ALTCOINS

4.1 Privacy

Under this title, we will discuss privacy problem of Bitcoin and altcoins in comparison with Bitcoin. Privacy is essential issue for cryptocurrency technology because of public blockchain that stores every transaction. While most of users want to have private transaction order to keep secret its transaction history, some of users have concerns about private decentralized transaction system because of some illegal trades like drug, gun, money laundering. We will present Bitcoin privacy and other privacy-centric alternative coins and other alternative coins, which has the same privacy problem as Bitcoin.

4.1.1 Bitcoin

- **Anonymity in Bitcoin**

It is very controversial that Bitcoin is anonymous or not. Firstly, it is need to be identified what is anonymity. Lexical meaning of anonymity is "being lack of identity". A Bitcoin address is hash of pubic key that is random alphanumerical string. This address does not give the real identity. In this context, Bitcoin has anonymity. But addresses are kind of pseudo identity which is kind of identity. This is called pseudonymity.

- **Privacy In Bitcoin**

In Bitcoin, user has addresses for transactions which pseudo-random strings called pseudonym addresses. But all transactions are publicly announced in blockchain. So, all history of an address is open to public including its bitcoin balance. It is maybe seemed that a user's transactions cannot be traceable but in fact most of addresses can be disclosed. Because users publish their addresses on their blogs or forums like BitcoinTalk, Reddit etc. Other users interact these people and this results a traceable environment in Bitcoin network. Thanks to this public system, an address can be

linked to user's real world identity. It means there is not unlinkability in Bitcoin. Reid and Harrigan studied on analysis of anonymity in the Bitcoin system especially on privacy [93]. It is hard to guarantee anonymity in the networked data which is open to public even if user uses pseudo names like in Twitter [93, 94]. In Bitcoin, this is similar to this research. In the research, it is shown that matching many public-keys with each other is possible with appropriate tools. Ron and Shamir analyzed full Bitcoin transaction graph in 2013 to understand statistical properties of Bitcoin graph [95]. Also, they analyzed users' behaviors for protecting their privacy.

Thus, Bitcoin has not privacy. As a first precaution, users change their addresses very often. But, all transactions are stored in blockchain. So, these transactions between addresses owned by the same user can be detectable. Also, with using new clustering heuristics like " if two (or more) public keys are used as inputs to the same transaction, then we say that they are controlled by the same user", addresses of a user can be detected [96]. A group of researchers found that an attacker with $2,000 budget could de-anonymize up to 60% of bitcoin clients on the network with finding links Bitcoin addresses to IP addresses [97].

Methods for Linking Addresses to User

1. **Shadow Addresses**

In Bitcoin system, value of an incoming transaction to user cannot be split. But if user needs to pay a smaller amount than incoming value, user needs to send the rest of fund to himself as "change". Lets say Alice has 50 BTC incoming transaction. It means Alice has 50 BTC. But, Alice needs to pay 20 BTC for an electronic device to merchant. So, Alice creates a transaction including 50 BTC incoming transaction to her as input value. But Alice wants to pay 20 BTC to merchant. Transaction fee 0.0002 is sent to miners. So, the rest amount 29.9998 BTC must be sent to her address as change. So, this address is generated by Bitcoin system as shadow address to prevent traceability of transaction of Alice [98].

In this case, this shadow address is firstly appeared in public log. So, it means that this new address is owned by Alice. Although this precaution, traceability of

addresses of Alice is possible at high probability. In current Bitcoin client, it is impossible that these transactions can be processed in two different transactions.

2. Multi-Input Transactions

Bitcoin user has incoming transactions as balance and spends them in future transactions. When amount of single transaction is not enough to pay for something, user needs to use more than one incoming transactions. So, inputs coming from different addresses in these transactions can be guessed as belonging to this user [98]. Lets assume that Alice wants to buy a good for 20 BTC. But, Alice has three incoming transactions on her different addresses values are 12 BTC, 6 BTC and 4 BTC. In the new transaction, Alice takes these 3 incoming transaction as input, sends the merchant address. Lets assume that transaction fee is 0.0002 BTC. Bitcoin client creates a shadow address for change. At the end of transaction, 20 BTC is sent to the merchant. 0.002 BTC is sent to miners for verification the transaction. Rest amount 1.9998 BTC is sent to new address of Alice as change. So, in this transaction, it is obvious that addresses of incoming transactions 12 BTC, 6 BTC, 4 BTC are owned by Alice. So, another way is found to detect other addresses of Alice. And now, in Bitcoin clients does not support different users to participate in single transaction. But new attempt for Bitcoin called CoinJoin provide multiple different users to have a single transaction together.

It is said that the first research about Bitcoin privacy is done by Androulaki et al [98]. According to their research held in the university, in a Bitcoin-like system, using these heuristics, 40% users can be linked to their bitcoin addresses even if they took precaution recommended by Bitcoin for their privacy [98].

Bitcoin recommends some precautions like using new addresses each time for getting new payment, not publishing addresses on public websites, hiding IP users' addresses using special tools like Tor (The Onion Routing), using some online mixing services [99].

Privacy is the one of the biggest concerns in Bitcoin. Thus, so many attempts have been emerged to prevent traceability of activities of addresses like mixing systems like Mixcoin [100], anonymization methods like CoinJoin [101].

- **Mixcoin**

Mixing is used to prevent traceability of transactions of users by mixing funds of users in the system. Mixing service takes all transactions and mixes them and gives these funds of transactions randomly to provide privacy to user. With mixing, original source of funds cannot be detected by others. Also, mixing services must be trustworthy. They cannot store logs of users' data and their system must be secure for not accessing of attackers to mixing service servers. Mixing service idea is shown in figure 4.1. Mixing services have 3 disadvantages:

1. Latency because of waiting for a large mix,
2. Mix can trace bitcoins,
3. There is no guarantee that mix gives back bitcoins of user (theft).

Figure 4.1 Mixing service principle.

Mixcoin is proposed by Bonneau et al. which is a mixing service for anonymous payments and fully compatible with Bitcoin [100]. Against active and passive attacker, Mixcoin offers anonymity. Privacy is important for financial transactions. So, cryptography community aimed at creating private payment methods without any central third party. Bitcoin is the first decentralized crytocurrency. But, it is not private. So, additional researches have been attempted to have private transactions in Bitcoin system. Mixcoin is one of them. Mixcoin has a mixing network system as an intermediary with randomized mixing fees which can be deployable to Bitcoin without any modification. Mixcoin uses standardized chunk size that is the amount

of money to divide into sending money. So, sender must divide his money into chunks to send them to mixing system in an order for each chunk. Sender sends his money to mix and takes back it after mixing without any fraud by mixing service. In Mixcoin, there are mixing parameters like below:

v: value (chunk size)to be mixed

t_1 **:** the deadline by which sender must send funds to the mix

t_2*:* the deadline by which mix must send funds to the sender

k_{out}*:* the address which sender aims to send his funds

p: the mixing fee rate sender will pay

n: a nonce used to determine payment of randomized mixing fees

w: the number of blocks the mix requires to confirm sender's payment

$\{data\}k_M$ *:* signed by private key of M (M : Mix)

Mixcoin protocol works like below:

1. Sender sends (v, $t_1, t_2, w, k_{out}, p, n$) to mix.

if M accepts terms:

2a. Mix sends $\{v, t_1, t_2, w, k_{esc}, k_{out}, p, n\}_{k_M}$ and M generates k_{esc}

if Sender pays on time:

3a. Sender transfers (v, k_{in}, k_{esc}) to by time t_1

if X=Beacon(t_1, w, n) X>p and M acts honestly

4a. M transfers (v, k'_{esc}, k_{out}) by the time t_2.

//PROTOCOL ENDS SUCCESFULLY

if X≤p :

4b. M retains funds.

if M steals funds:

4c. There is no transfer to k_{out} by the time t_2.

Sender detects theft after t_2 and publishes $\{v, t_1, t_2, w, k_{esc}, k_{out}, p, n\}_{k_M}$.

if Mix rejects terms:

2b. M sends rejection. Sender destroys k_{out}.

if Sender does not pay:

3b. M aborts protocol.

- **Tor For Bitcoin Privacy**

Tor is used for anonymous communication over the Internet. It enables untraceable communication between user and destination without any network surveillance or traffic analysis. It is free software that enables low-latency anonymity network and uses The Onion Router base. It is based on volunteer network community. Tor uses onion-like layer system that prevents traceability of user activity by others. Onion routing is implemented by encryption in the application layer of a communication protocol stack like onion style. It has 3 Tor relays that are called *Guard, Middle, Exit*. User keeps secret own IP address in the network. Tor network structure is shown in figure 4.2.

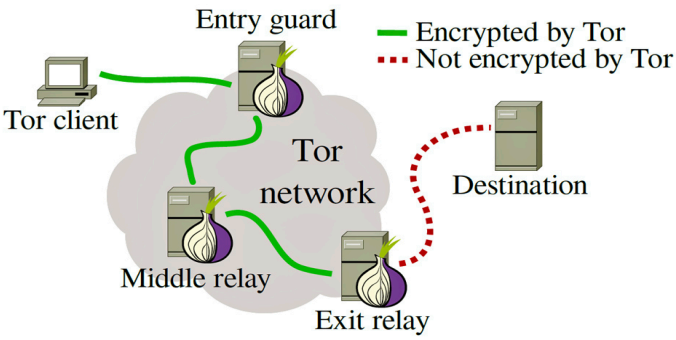

Figure 4.2 Tor structure [102].

- **Benefits of Tor for Bitcoin Privacy**

Thanks to Tor, IP of user cannot be traceable while user sends transaction in Bitcoin network. Tor hides IPs and locations using relays by encrypting each connection between relay peers. Bitcoin privacy increases with using Tor. But, there are some countermeasures of using Tor in Bitcoin. Tor helps privacy of Bitcoin but it is not perfect for privacy.

- **Disadvantages of Tor for Bitcoin Privacy**

Biryukov and Pustogarov claimed that Bitcoin over Tor is not a good idea although Bitcoin developers recommend using anonymization tools like Tor for privacy [103].

To identify transactions of user, it needs two steps:

1. Linking transactions to the user IP address.
2. Finding link between user all addresses and user's transactions.

Although, user use mixing services, it is still vulnerable to be detected via IP address leakage. In case of IP address traceability, Using Tor or VPN (Virtual Private Networks) is recommended by Bitcoin community. SPV clients are vulnerable to spoofing and MITM (man-in-the-middle) attacks if they do not use a secure channel like SSL (Secure Socket Layer) [104, 105]. With BIP70, proposed by Gavin Andresen and Mike Hearn, Bitcoin clients became more resistant against to MITM attack [106]. Tor prevents DDOS (Distributed-Denial-Of-Service) attacks on clients. Biryukov and Pustogarov claimed that Tor causes MITM attacks to users. Their other finding is fingerprinting technique that is about *address cookie* on the user's computer. For same users, different sessions can be established even if user wants to connect to the Bitcoin network, cookie is still valid and reveals user IP address [103].After attack, Tor network is changed like in figure 4.3.

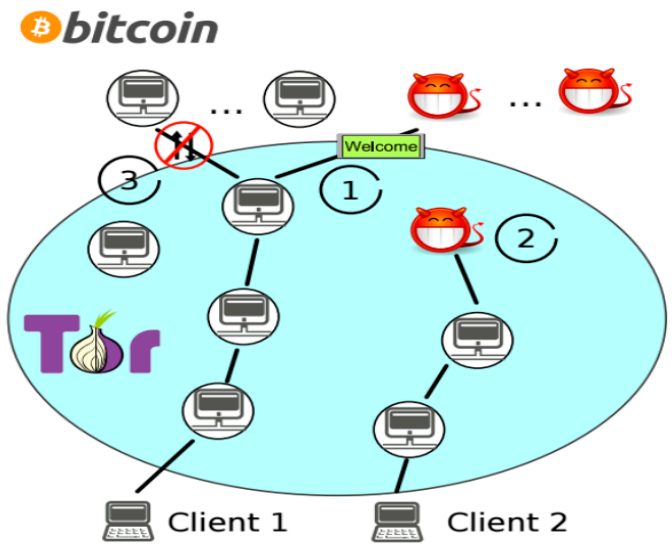

Figure 4.3 Tor network after attack [103].

4.1.2 Dash

Dash is the first altcoin that focused on privacy especially. Dash proposes strong anonymous cryptocurrency experience [107]. Also, Dash is assumed as Bitcoin's cousin and it boomed in the first month after launched [108]. SharedCoins, Dark Wallet, CoinShuffle, JoinMarket also use CoinJoin idea.

- **CoinJoin**

Gregory Maxwell proposed CoinJoin as an anonymization method for Bitcoin in 22 August 2013. CoinJoin depends on joint transactions idea. If a user wants to transfer funds to someone else that user must find another user who is about to send payment to someone else to make a joint payment together. In a transaction, two payments are handled [101]. Joint transaction fees are reduced because number of transaction is reduced. But, Coin Join must be implemented very carefully. Because link between input and out of transaction can be detectable. Also, CoinJoin has two important weaknesses. It must handle all signatures in case of leak. Also, CoinJoin transactions can be traced by amounts, through change linking. In figure 4.4, a CoinJoin transaction shown.

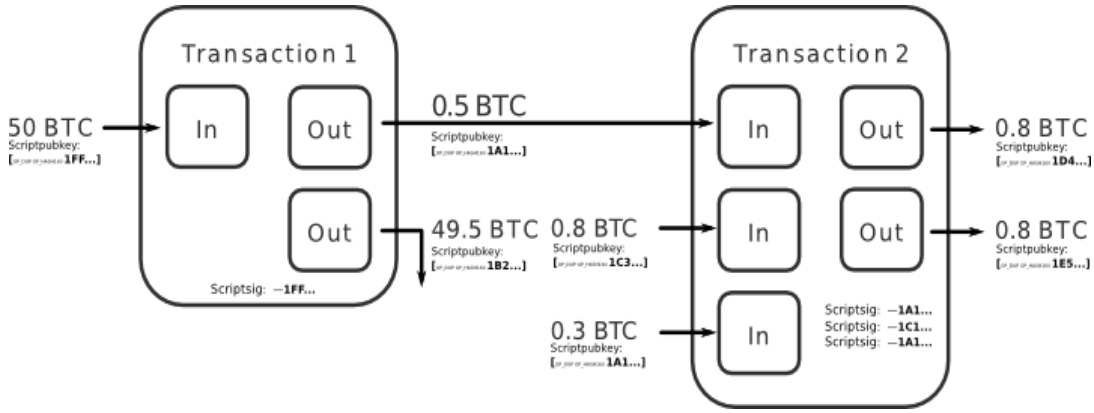

Figure 4.4 Joint transaction in CoinJoin [101].

Dash uses Darksend coin mixing service that makes use of Coin Join idea. Unlike Coin Join, Darksend uses more than one mixing steps called chaining approach to make traceability harder.

4.1.3 ZCash

While in Bitcoin transactions origin, destination and amount of transaction are published on the blockchain publicly, Zcash keeps secret all these information from others, but Zcash proves that inputs and outputs of the transaction are valid without conveying any additional information about the transaction. Zcash is not yet used as actual money and it is only a testnet to detect all bugs before it will appeared on the real market. Now, Zcash is highly experimental and has risks for user but it is planned that Zcash will be launched in six months [104, 109].

Bitcoin has privacy problem despite of using with mixing services which are not reliable third parties, whereas Zcash proposes a privacy-preserving system. Although, Zcash was not experienced yet in the real market, it is a promising full-fledged altcoin for privacy concerns. Zcash investor, Bo Shen indicated that privacy is a very important challenge with current blockchain technology, but existing solutions like private blockchains do not really solve the problem. The technology behind Zcash does[15] as seen in table 4.1. Roger Ver, another investor of Zcash believed in Zcash privacy preserving system is better than Bitcoin with saying that Zcash is strongest where Bitcoin is weakest; It gives users the easy ability to maintain their financial privacy.

Table 4.1 Bitcoin and other anonymity applications comparison [74].

Coin System	**Used Type**	**Anonymity Attacks**	**Deployability**
Bitcoin	Pseudonymous	Transaction graph analysis	Default
Manual Mixing	Mix	Transaction graph analysis/badmixes, peers	Usable today
CoinJoin	Mix	Side channels, bad mixes	Bitcoin-compatible
Zerocoin	Cryptographic mix	Side channels(possibly)	altcoin
Zcash	Untraceable	None known	altcoin

[15] Zcash website https://z.cash/ Retrieved 27 February 2016

4.1.4 Ripple and Peercoin

Anonymity is not a design aim of Ripple. Its privacy is the same as Bitcoin architecture. Accounts' balances are open to public. But it has untraceable transactions if sender or receiver or gateway does not disclose transactions. User connects gateway to exchange their funds. Only gateway, sender and receiver know the transaction detail. Thus, Ripple does not have traceable transactions. Also, Ripple has anonymity because of using nicknames and a transaction cannot be linked to real owner unless it is disclosed by owner, sender or gateway.

Peercoin is similar to Bitcoin in terms of anonymity and privacy. Peercoin uses blockchain to store all previous transactions by disclosing sender and receiver addresses, balance. Users have alphanumerical addresses to send and receive peercoins and all transactions are open to public. By tracing transactions, there is a high probability to identify real identity. Also, if user buys something by using peercoins from a merchant, merchant must know customer's address and identity to send goods. Hereby, merchant can check all previous and future transactions of the merchant unless customer stops using this address. So, it is pseudonymous and has traceable transactions. There is linkability between transactions and real identity of user by investigating transaction activity, activity time etc.

4.1.5 Side Effects of Privacy/Anonymity

While privacy is a major property for users, a private currency can be used for money laundering or in darknet market like Silk Road to buy illegal drugs [110]. Silk Road operates as a Tor hidden service and uses Bitcoin as exchange currency for illegal transactions. Silk Road was used for narcotics trafficking, computer hacking, money laundering [111]. Bitcoin price fell down from $145.70 to $109.76 after Silk Road closure in 2013. Silk Road creator, Ross Ulbricht said that Silk Road would not have been possible without Bitcoin in an interview [112]. So, Bitcoin had bad reputation in the market. Silk Road provided share 4.5% of the entire Bitcoin economy [113].

Money laundering means that gain of crime is converted into legitimate money. Although Bitcoin transactions are publicly available in blockchain, thanks to some mixing and anonymization tools like Dark Wallet can make money laundering possible. Cody Wilson, one of the inventors of Dark Wallet stated that Dark Wallet is just money laundering software [114]. Also, Bitcoin is not monitored by financial authorities. So, Bitcoin can be a good exchange method for money laundering with some appropriate tools. Some anonymization services like Bitcoin Fog, BitLaundry, Send Shared were analyzed and researcher found that Bitcoin Fog and Send Shared were successful at making transaction anonymized but they found link between inputs and outputs in BitLaundry [115]. Like these attempts cannot be a good choice for money laundering. For example, in January 2016, 10 people were arrested because of accusation of money laundering over Bitcoin [116]. On the other hand, UK Government published a report which indicates digital currencies posing lowest risk for money laundering [117].

4.2 Transaction Malleability

Under this title, we will present transaction malleability problem in Bitcoin and other altcoins. Transaction malleability means that a transaction can be changeable after sender sends to the receiver by malicious third party. There is a simple explanation of transaction malleability with a basic example.

Basic Example Of Transaction Malleability:

1. Alice wants to send some funds to Bob and creates a transaction.
2. Lets assume that transaction data: 12 HASH(123)=173af653133d964edfc16cafe0aba33c8f500a07f3ba3f81943916910c257705= TXID__OLD (TXID__OLD stands for transaction id before changed by attacker.)
3. Alice publishes this transaction on the Bitcoin network.
4. Attacker, Evil takes the transaction and modifies it with putting a zero at the start. Again, 0123 is equal to 123. So, without invalidating transaction, a small modification will result a different hash like below. Because, even if one bit is changed in data, result will be very different.

HASH(0123)=
4a6d4875f4d20332e6c5251f484a1d305818bcf37f1e3829c50b97b1068dba0c
= TXID_NEW

5. Attacker makes this transaction with TXID_NEW in Bitcoin network where TXID_NEW represent transaction id after change by the attacker.
6. After validating this transaction, payment goes to Bob. But Bob does not know that he receives the payment by Alice. Because Bob only checks if TXID_OLD is appeared and validated on the blockchain. But Bob cannot find TXID_OLD. Then, he decides that Alice did not pay to him. And only one transaction is confirmed by Bitcoin system. In attack, the aim is to make TXID_NEW validated before TXID_OLD.

Until here, it seems as if it is a simple problem that does not harm anyone seriously. But it is still problem. If we consider another case like Bob is the attacker, this case can be more harmful to Alice. Recall that attacker changed the TXID and Bob thought that Alice did not send the payment. If Bob is the attacker, Bob says Alice to resend the payment. Because Alice's protocol checks TXID_OLD and realizes that TXID_OLD does not appeared on the blockchain. So, Alice's protocol resends the payment to Bob again and again. In this situation, Bob, the attacker takes too many payments and Alice cannot realize what is going on. This attack is called as transaction malleability.

Transaction malleability affects chained transactions. Because some Bitcoin users accept the transaction less than 6 confirmations. So, if user pays to another with non-confirmed transaction as input to his transactions, this transaction can be invalidated after a while. Because attacker changes the transaction id coming to him and new transaction id is confirmed before the original one. Then, incoming transaction remains invalidated and his next transaction created by using this incoming transaction will be invalid. Previous invalid transaction will affect the next one. If the chain depended on the first invalid transaction is longer, the more number of transaction will be invalidated at invalidation of the first one like domino chain.

Reason of transaction malleability is not to cover all data while signing transaction. So, uncovered data can be changed in a simple way. So, result hash is changed.

4.2.1 Bitcoin

Transaction malleability was a type of bug in Bitcoin protocol. It is firstly defined in 2011 [118]. It allows changing transaction id (TXID) without invalidating transaction like without damaging signature. Each transaction is identified with only one TXID. Because all transaction information is hashed with double SHA-256 and a unique serial number is found. But transaction malleability makes possible a transaction to have more than one TXID. Signature locates in script (data part of transaction). Signature does not cover redeem script. The reason is that if signature covers script, it means to sign the signature itself that is apparently impossible. So, attacker can make trivial modification like zero-padded push operations in script and all data including script part is hashed with double SHA-256 and as a result a different hash is created. This hash represents transaction id. So, this modification means that attacker can change transaction id without invalidating transaction. And if a transaction is just checked with its TXID whether receiver takes the payment or not. So, problem continue from this point. While receiver is waiting for the real transaction with real TXID, attacker changes the real TXID and make this transaction validated before the real transaction TXID. In fact, transaction is sent to the receiver but the receiver does not know that he takes the payment. Because receiver cannot see the real TXID (supposed one) on the blockchain. Because receiver protocol just checks TXID. So, he decides that payment is not paid himself.

There is another vulnerability for transaction malleability. Signature is vulnerable because of OpenSSL accepts different encodings of signatures. Nine types of malleability are posted in Github by Pieter Wuille for BIP 62 (Bitcoin Improvement Proposal 62) [116, 119]. These are shown below:

1. ECDSA signature malleability: Two points are calculated according to ECDSA algorithm. They are two points on elliptic curve. It is easy to find another set of points using signature that these points encode the same point on the elliptic curve.
2. Non-DER encoded ECDSA signatures malleability: OpenSSL accepts more than one standarts. So, with v0.8.0 non-DER signatures re not allowed to use anymore.

3. Non-push operations in scriptSig malleability: If any sequence of script operations in scriptSig is utilized, it is still results same data pushes.
4. Push operations in scriptSig of non-standard size type malleability: Same push operation can be handled with two different push commands by preserving validity of transaction. For example; OP_PUSHDATA1, OP_PUSHDATA2, OP_PUSHDATA4. This results transaction id change.
5. Zero-padded number pushes malleability: With some zero padding instead of interpreted numbers by scripPubkey opcodes.
6. Superfluous scriptSig operations malleability: Extra data push to start of the script which are unused causes malleabiliy.
7. Inputs ignored by scripts malleability: Some opcodes like OP_DROP the last data push is dropped.
8. Sighash flags based masking malleability: In signing operation, some data can be ignored because of some sighash flags. This type of malleability cannot be fixed with extra consensus rules.
9. New signatures by the sender malleability: It is allowed to create new signatures with same inputs and outputs if user has same private key. This type of malleability cannot be fixed with extra consensus rules. Others generally can be fixed.

Firstly, we give a basic explanation of transaction malleability like below. After we will explain a technical detail of push operations in scriptSig of non-standard size type malleability.

- **Technical Detail of Transaction Malleability**

To give technical detail of transaction malleability, we choose a transaction malleability type from list above. And, we choose a transaction from Bitcointalk declared by owner that there is something wrong with the transaction. Owner indicated that he received a mail by blockchain declaring his transaction is

invalidated so removed from their database [16]. Table 4.2 shows the transaction data parts.

And also, owner stated that there is a warning in his transaction page on blockchain.info like " *Warning! this transaction is a double spend of 112591137. You should be extremely careful when trusting any transactions to/from this sender*.

This transaction id was c23f...But he realized that txid was changed to this one ed5d1...

We analyze the second transaction id, changed version. Because the first one, produced by the owner one, unchanged one was removed by blockchain.info database. In transaction malleability attacks, modified one is always confirmed before the first one to create a confusion to receiver. So, we could find the confirmed version of transaction on blockchain database. That is the reason why we could not and analyze the original one. We benefit from Ken Shirriff's blog post while analyzing the transaction [120].

Raw data of the modified version of transaction [121]:


```
{
"ver":1,
"inputs":[        {
"sequence":4294967295,
"prev_out":{
"spent":true,
"tx_index":50231614,
"type":0,
"addr":"135RprcM8sqVedteWJtwSwSGEJ7BBMXers",
"value":20800000,
"n":5,
"script":"76a91416c6a36b6c6160dff9bf4d498430e9bf06be534e8
8ac"          },

"script":"4d48003045022100e79b5f1de9bcdbdf1d5abf978fb3da1
c3b2aed3a1cca6c767ff71d65e425fc56022014820f74cea6f78851a1
d105c90b85f61d95b320a1d92efbc2da133c3a175db9014d210002ef4
f38c8e43dee5e089cbd08e8969412adfb87c2fdc847cd0a45cc9980e4
4e76"
 }
```


[16] https://bitcointalk.org/index.php?topic=459499.0

In this raw transaction, the red parts are changed by the attacker. Because normally, in Bitcoin transaction, scripts do not start with "4d". They start with number of bytes of pushed data.

Table 4.2 Bitcoin transaction partitions.

PUSHDATA 48 (after modification by attacker OP_PUSHDATA2 0048)		48 (after modification by attacker 4d 48 00)
signature (DER)	sequence	30
	integer	02
	length	21
	X	00e79b5f1de9bcdbdf1d5abf978fb3da1c3b2aed3a1cca6c767ff71d65e425fc
	integer	56
	length	02
	Y	14820f74cea6f78851a1d105c90b85f61d95b320a1d92efbc2da133c3a175db9
SIGHASH_ALL		01
PUSHDATA 21 (after modification by attacker OP_PUSHDATA2 0021)		21 (after modification by attacker 4d 21 00)
public key	type	02
	X	ef4f38c8e43dee5e089cbd08e8969412adfb87c2fdc847cd0a45cc9980e44e76

The way of modification is simple. Because this modification does not change signature. But transaction hash is changed. Again, transaction reaches its destination to receiver. If receiver's protocol just checks the transaction id to understand whether it receives transaction or not and protocol accepts as it does not receive the

transaction because of changed transaction id. It searches for the unmodified transaction id.

Technically, attacker just change the push data parts by adding "4d #BytesofData 00" and "OP_PUSHDATA2 00 #BytesofData ". Before modification, it is written just 48. It means to push (hex) 48 bytes of data.

After modification, it is written 4d 48 00 and OP_PUSHDATA2 004. It means to push the next two bytes (48 00) are the number of bytes to push. It takes input in reverse order like OP_PUSHDATA2 00 48. So, it means that push 48 bytes of data that has same meaning before modification.

As a result, both of them have same meaning like pushing 48 bytes of data. But attacker makes use of this bug and causes changing of transaction id. Thus, receiver cannot understand if the transaction reaches to him, if he only checks transaction id.

Transaction malleability is considered a special type of double spending attack by Decker and Watterhofer [122]. In double spending, with the same fund, attacker tries to buy two different goods. But, transaction malleability attacker modifies incoming transaction with changing transaction id to create confusion to sender with aiming resending transaction from sender.

- **MT.Gox Incident**

MT.Gox was one of the most popular Bitcoin exchange site established in July 2010 in Tokyo, Japan. It held nearly 70% of Bitcoin transaction by 2013 and increased its market share nearly 90% by 2014 [123,124]. In 10 April 2013, Bitcoin price dropped drastically from $266 to $100 in a few hours. Because it could not handle high volume trades. Also, Mt.Gox faced DDOS attacks many times. On 7 February 2014, Mt.Gox announced a technical issue, stopped BTC withdrawals and promised to have an update in 3 days. Bitcoin price dropped 10% because of this news. After 3 days, Mt.Gox did not publish an update and did not open withdrawals and stated the issue as transaction malleability. This time, it was different than other crashes. There was a serious problem rooted from transaction malleability. Mt.Gox was using output-based system that it checks only transaction ids while using incoming outputs

as inputs of its new transactions. On 24 February 2014, Mt.Gox website went offline and stopped all trading after losing 850,000 bitcoins cost $460 million with theft which could not be detected for years [125]. Mt.Gox lost its customers bitcoins and could not manage to give them back to owners. Finally, Mt.Gox was closed on 25 February 2014 after huge bankruptcy. Now, there is a series of announcement about bankruptcy on its website [17]. Bitcoin price dropped from $950 to $550 USD after Mt.Gox incident.

Transaction malleability had great attention in MT.Gox incident. Because there was a huge bitcoin loss due to a bug in Bitcoin protocol of Mt.Gox. To fix signature malleability, Bitcoin developer team published BIP-66 as a soft fork to provide strict DER (Distinguished Encoding Rules) signatures [126]. Also, BIP-62 was developed for fixing transaction malleability bug [119].

Decker and Wattenhofer claimed that Mt.Gox did not lose 850,000 bitcoins just because of transaction malleability attack. In their study, they investigated the incident and asserted that nearly 386 bitcoin were stolen using malleability attack. Thus, they have questions about rest 849,600 to Mt.Gox directors [122]. But, Bitcoin community claimed that they could not find all changed transaction for malleability attack. Because invalid transactions are discarded by the system. So, researchers could not get all data about Mt.Gox incident [127].

Before Mt.Gox incident, in 2013 Andrychowicz et al. published a paper from University of Warsawabout how to tackle with transaction malleability bug in Bitcoin protocols [128]. They proposed Bitcoin-based timed commitment scheme and *Fuse* transactions to have non-malleable transactions.

4.2.2 Zcash

Zcash claims that Zerocash protocol is transaction non-malleable [5]. Zcash transaction non-malleability scheme prevents attacker to change transaction before adding to the blockchain. They proposed a scheme to keep pour transaction ids

[17] For more information: https://www.mtgox.com/

unchangeable formalized as TR-NM. It is not fully tested in real market yet. So, we cannot know exactly if Zcash will face transaction malleability issue in the future.

4.2.3 Dash, Ripple, Peercoin

Dash code is forked from Bitcoin code. Dash did not face transaction malleability problem till now. Also, transaction malleability depends on implementation in exchange clients. After Bitcoin transaction malleability incident, other cryptocurrencies fixed this bug in their software.

Ripple utilizes different method for handling transaction malleability. Its transaction has an inner transaction that is signed. Inner transaction contains all necessary data about transaction like recipient, amount. But there is still a problem that a signature cannot be signed. So, Ripple advises that a signature must have only one correct form.

Peercoin uses Bitcoin code. So, when Bitcoin was affected by Mt.Gox incident, Peercoin was vulnerable, too. After Bitcoin fixed bug, Peercoin adopted the Bitcoin fix. There is any peercoin theft reported because of transaction malleability. Because Peercoin was not a high price to steal.

4.3 Scalability

Under this title, we will present scalability problem in Bitcoin and some altcoins. Scalability is a prominent problem in peer-to-peer decentralized system. Because with time, network and hardware constrains come as problems and solutions to these problems lead new scalability problems. We will discuss some solutions for scalability problem in Bitcoin and discuss scalability solutions of other altcoins.

4.3.1 Bitcoin

Bitcoin is based on peer-to-peer network to distribute all information to all peers in Bitcoin network. Bitcoin network is becoming larger day by day because of increasing its popularity in the market. Thus, Bitcoin network have trouble with scalability issue like network size, storage requirements and network bandwidth.

At the time of writing, Bitcoin network has nearly 240000 transactions per day, 400,000 unique Bitcoin address, 57902 MB blockchain size, 0.7 MB block size, nearly 1500 transactions per block[18]. Bitcoin block size is fixed to 1 MB by Satoshi Nakamoto to avoid denial-of-service (DoS) attacks on the Bitcoin network. Also, block size limit can be a necessity to avoid centralization of the Bitcoin network. But with increasing number of transactions, block size needs to be larger to validate more transactions in a block in terms of making validation faster. Because if a block cannot include all transactions need to be validated because of its 1MB limit, rest transactions waits for another block. And, every block is mined in 10 minutes in average as a rule decided by Bitcoin creator Satoshi Nakamoto [3]. Bitcoin network has 7 tps (transactions per seconds due to the 1MB block size limit, while VISA network has ability to handle 8500 tps for payments and PayPal has 100 tps. Bitcoin is getting popular as an alternative to the banking model because of its decentralized structure. So, Bitcoin network needs to have greater than 7 tps system to serve to more users. So, it needs to increase its 1 MB block size limit. But if Bitcoin protocol allows using more than 1 MB block size, blockchain size increases and it gets harder to download and verify all blocks from the genesis block for a full node with a greater blockchain size. This is another scalability problem for the network. Also, with increasing number of transactions network power must be increased to distribute all new transactions and confirmed transactions.

- **Block size Scalability Problem**

Number transactions of Bitcoin users is increasing day by day. A block includes a limited number of transactions because of 1MB block size rule. So, some transactions wait for the next block to be confirmed by causing latency in transactions. So, if the blocks aren't big enough, transaction fees will rise to be first to be confirmed by miners. If block sizes are too big, number of miners who has such high technical limitation increases. So, miners become centralized. Or, bigger blocks take more time for propagation. Longer propagation time causes high orphan rate. Because miner mines on older blocks while new blocks are propagating. Also,

[18] For more information: https://blockchain.info/charts

Chinese mining pools which consist of 60% of the network hash rate do not want bigger blocks like 20 MB because of the China network bandwidth [129]. For this issue, there are BIP (Bitcoin Improvement Proposal) proposals by Bitcoin developers and researchers.

- **Voting For Block Size**

There were lots of proposals for block size. Bitcoin community could not have consensus on block size. So, they decided to ask all Bitcoin miners. Every miner encoded *'BV'+BlockSizeRequestValue* in blocks. All votes in blocks can be found at blocktrail.com [130]. This voting system is debatable. Because large mining pools can increase block size and small mining pools have trouble with handling large block size.

1. **Default**

   Block size is fixed to 1 MB defined by Satoshi Nakamoto at the first of Bitcoin project. Miners choose default, if they want block size to be fixed to 1MB or have any other preference [131]. At the time of writing, default has 381742 votes [132].

2. **Removal of The Block Size Limit**

   Some part of Bitcoin community wants to remove block size limit. Blocks are mined at any size. But, block size limit is what ensures everyone can participate in the Bitcoin network. Because, unlimited block result big size blocks and these are hard to handle for small pools. Big pools get the all control of Bitcoin network and Bitcoin becomes centralized. This result contradicts the idea of Bitcoin.

3. **BIP-100**

   Jeff Garzik, Bitcoin Core Developer, proposed to change static 1 MB to dynamic block size decided by miners in the future. Every 3 months the limit changes at max twofold bigger or smaller decided by miners voting. Upper limit is fixed to 32 MB. This change is decided as a hard fork. Miners vote for their

decision, then most common minimum value is chosen by the system. At the time of writing, BIP 100 has 13148 votes [132].

4. BIP-101 (BitcoinXT)

Gavin Andresen, Chief Scientist of the Bitcoin Foundation & Bitcoin Core Developer, proposed to increase block size limit to 8 MB by doubling every two years decided for twenty years. First change of Bitcoin system is decided as a hard fork and future doubling will not be hard fork. Because in the first hard fork, future change is already decided. At the time of writing, BIP-101 has 56 votes [132].

5. BIP-102

This is alternative to BIP-100 proposed by Jeff Garzik as a fallback in case of not reaching in time of BIP-100 and BIP-101. It proposes 1 MB block size to 2 MB block size with a hard fork.

6. BIP-103 Pieter Wuille's Proposal (Segregated Witness)

Pieter Wuille, Bitcoin Core Developer, proposed increasing the block size limit by 4.4 percent about every 97 days. This means that an annual block size limit increases at 17.7% with a hard fork.

7. 8 MB

Block size is fixed to 8MB. There is no specific BIP code. At the time of writing, 8 MB has 5667 votes [132].

8. Adam Back's Proposal

Hashcash inventor and Blockstream President, Adam Back proposed a change to Bitcoin block size increasing 2 MB immediately, 4 MB after two years, 8 MB after four years and staying flat at 8 MB till the end [133].

- **Hard Fork or Soft Fork for Block Size Change?**

Even if community has a decision on consensus about new block size, there is still debate about change if it is applied as a hard fork or soft fork. Peter Todd claims that soft fork is safer than hard fork. Soft fork is a kind of modification on Bitcoin protocol by adding new rules to the protocol, making previously valid blocks invalid. Hard fork is kind of modification on Bitcoin protocol by deleting rules from the protocol, making previously invalid blocks valid. So, Peter Todd asserts that hard fork is less safe because of being dangerous of making invalid blocks valid [79]. Also, in Bitcoin blog it is declared that contentious hard forks are harmful for Bitcoin [134].

On the other hand, Mike Hearn, Former Bitcoin Core Developer says that soft forks are bad [135]. Because miner mines on a block that miner thinks the block is valid, after miner learns that the block is ignored by others. And miner tries again and new block is also ignored. Until miner hears about rule change, he wastes too much energy. So, soft fork is not good.

Final Decision: Consensus About Block Size

After a long discussion, In 21 February 2016, in Hong Kong's Cyberport, Bitcoin development community members and people from Bitcoin industry agreed on a consensus about block size. They chose Pieter Wuille's Segregated Witness (SegWit). They decided on points below [136]:

- *We understand that SegWit continues to be developed actively as a soft-fork and is likely to proceed towards release over the next two months, as originally scheduled.*

- *We will continue to work with the entire Bitcoin protocol development community to develop, in public, a safe hard-fork based on the improvements in SegWit. The Bitcoin Core contributors present at the Bitcoin Roundtable will have an implementation of such a hard-fork available as a recommendation to Bitcoin Core within three months after the release of SegWit.*

- *This hard-fork is expected to include features which are currently being discussed within technical communities, including an increase in the non-witness data to be around 2 MB, with the total size no more than 4 MB, and will only be adopted with broad support across the entire Bitcoin community.*
- *We will run a SegWit release in production by the time such a hard-fork is released in a version of Bitcoin Core.*
- *We will only run Bitcoin Core-compatible consensus systems, eventually containing both SegWit and the hard-fork, in production, for the foreseeable future.*
- *We are committed to scaling technologies which use block space more efficiently, such as Schnorr multisig.*

Based on the above points, the timeline will likely follow the below dates.

- *SegWit is expected to be released in April 2016.*
- *The code for the hard-fork will therefore be available by July 2016.*
- *If there is strong community support, the hard-fork activation will likely happen around July 2017.*

- **Blockchain Scalability Problem**

According to Visa website, Visa network is capable of handling 24000 tps [19]. If Bitcoin network handled such a big amount of transaction with 250 bytes of bitcoin transaction size, block size would be 3. 5 GB and blockchain would increase 190 TB per year. This amount of blockchain size cannot be handled by a regular home desktop and it is impossible to set up such a great bandwidth for regular devices. Downloading such big data is very hard. So, number of full nodes decreases and

[19] For more information: https://usa.visa.com/run-your-business/small-business-tools/retail.html

miners have difficulty in mining on 3. 5 GB block because of hashing operations. Thus, number of miners decreases and small number of miners start to behave a centralized system and this is not the desired design principle of Bitcoin.

Block pruning might be a solution for blockchain size problem for full nodes. Technically, it means that last 550 blocks of blockchain are stored on user's storage to confirm transactions. But pruning is not a final solution. Because user needs to download all blockchain as a first step, then pruning can be made. Thus pruning is not still a good solution.

- **Network Scalability Problem**

If block size is increased, some network problem starts to happen to handle block on the network while distributing the network. Poon and Dryja proposed Lightning Network that proposes network of micropayment channels to send transactions [137]. Lightning Network aims to solve the scalability issue by implementing hashed timelock contracts between users. Lightning Network promises that all world transactions including 7 billion people can be handle with LN by using 133 MB blocks presuming 500 bytes transaction size and 52560 transactions per year. Also, regular current desktops can handle being a full node storing blockchain size by pruning out blockchain to 2 TB storage size [137]. Peter Todd declared that Lightning Network allows scalability Bitcoin system without diminishing security [138]. Zohar and Sompolinsky proposed GHOST (The Greedy Heaviest-Observed Sub-Tree) protocol that uses off-chain blocks to make Bitcoin system more scalable and secure [139].

- **A Scalable Blockchain Protocol Proposal: Bitcoin-NG**

Eyal et. al. proposed next-generation blockchain protocol called Bitcoin-NG to handle scalability problem in Bitcoin [140] . They developed Bitcoin-NG blockchain protocol by allowing Bitcoin network to handle the highest throughput possible in Bitcoin network. Also, it enables lower transaction latencies. It allows transactions to get confirmation in seconds instead of minutes without any changing in Bitcoin architecture.

Scalability is a big challenge for its future. There are two choices to tackle scalability problem which one is increasing block size like proposed in BIPs and another one is to decrease block intervals. But both of them will result in different problems. They both cause forks. Forks are not secure for Bitcoin system. Because there are more than one branch in blockchain and this results undecided situation. Except one fork, rest forks need to be pruned. Mining power secure Bitcoin. It means that mining power for pruned branches is wasted. So, it gives opportunity attackers successful attacks like 51% attack or selfish mining because wasting mining power. Also forks reduce fairness. Because Bitcoin network compensates miners according to their power. When forks occur, less powerful miners are affected more than more powerful miners by earning less proportional to their power. So, they decide to join larger pools. Thus, this fairness causes centralization in Bitcoin network.

Bitcoin-NG proposes scaling without limits. In Bitcoin, a block creates in 10 minutes by miners. But Bitcoin-NG selects a leader to investigate future transaction as soon as possible after they are created. In every epoch, a leader is responsible for checking transaction until another leader is chosen. It has two different blocks called microblock and keyblock. Keyblocks are utilized for choosing a leader and generated by mining with proof-of-work in 10 minutes like in Bitcoin current system. Every keyblock is start of new epoch. Microblock contains transactions and generated by leader of epoch. They are not created with proof-of-work. But they signed with epoch leader's private key. Keyblocks contains only coinbase transactions. So, they are small-sized. Miners cannot get leadership easily because of needing proof-of-work. Lead miner can create microblocks by simply signing. Figure 4.5 shows Bitcoin-NG blockchain structure.

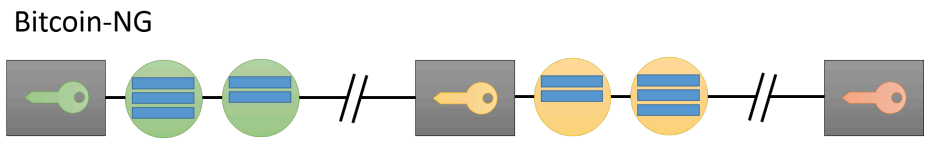

Figure 4.5 Bitcoin-NG blockchain structure (square ones:key blocks, round ones: microblocks)[156].

After their Bitcoin-NG experiment with 1000 nodes, they got significant improvement in terms of performance and fairness with optimal scaling. Bitcoin-NG

scalability can be limited only by individual node limits and network physical properties.

4.3.2 Ripple

Ripple is designed to scale high number of transaction per second. Bitcoin can handle 7 tps. Ripple uses off-chain transactions and handles 1000 transactions per second [141]. If validators, which cannot keep up with the network transaction volume, go out from the network, validation network get smaller and this causes less secure system. Because small network can be owned by a malicious third party and it can manipulate transactions. If there is insufficient validation network, Ripple network refuses transactions for security of users. Ripple network protects itself from being too small size. So, transaction fee is increased by validators in case of being under load. This causes that only the most valuable transactions are processed. Ripple transactions are so small that validators handle it with normal Internet bandwidth. Small transactions do not create big blocks and large blockchain size. Also, transaction processing can be done in parallel. So this makes Ripple highly scalable So, Ripple has less scalability problem in contrast to Bitcoin. Ripple has dynamic scalability system. If network is flooded by transactions, Ripple protocol automatically increases transaction fee so that attacker needs high budgets and finally cannot be successful at flooding the Ripple network.

Ripple does not use old transactions to process new transactions. Thus, Ripple does not require to store the whole history of transactions. So, it does not have blockchain scalability problem. Ripple uses an iterative consensus process that transaction is confirmed faster than Bitcoin and more energy efficient. David Schwartz, Ripple developer indicated that Ripple is designed to be a highly scalable system [142].

Generally for transactions, the most expensive step is validation of signatures. Ripple creates Ripple clusters to distribute work of validation to Ripple servers. If this cluster validates signatures, other servers do not need to spend power for validation again. This decrease work load and increase effective work in the network.

Bitcoin and Ripple has similar limits of scalability. But Ripple already has some more scalability features than Bitcoin. But Bitcoin has ability to add new scalability features to the system easily and some developments are still in progress [143].

Scalability is not easy problem to solve for cryptocurrency. While increasing transaction volume, block and blockchain size scalability problem raises.

4.3.3 Zcash, Dash, Peercoin

Zcash is not on the real market, yet. It is still in testnet. Developers did not decide on the block size. They indicated that there is a probability to use one of the block size proposals for Bitcoin in the future.

Dash utilizes incentivized masternode idea that carry load of the network by allowing millions of transaction per day. Also, Dash uses limiting masternode number in the network. Limit is determined by number of coins exist in the network. If there are 5.5 millions of Dash in the network, 5500 masternode should be in the network because 1000 Dash must be owned by each masternode when they are accepted to the network.

Table 4.3 Incentivizing masternode [144].

Required DASH per Masternode	DASH Price	Revenue Per Node / Cost Per Month / Dash Per Day	Network Actions Per Second	Network Storage / GB per [2]	Masternode Count [3]
1000 DASH	$10	$160 / $20 / 0.6	275,000 [1]	5.5 TB / 10 GB	2750
500 DASH	$100	$800 / $100 / 0.3	1,000,000 [4]	44 TB / 80 GB	5500
250 DASH	$1000	$4100 / $400 / 0.15	4,000,000 [5]	176 TB / 320 GB	11,000

1. 10ms per action, a quorum size of five and 50% utilization for other tasks
2. Average redundancy of 5x
3. 50% of all coins used for Masternodes
4. 5ms per action, a quorum size of five and 50% utilization for other tasks

Peercoin scalability is same as Bitcoin scalability problem. Because Peercoin uses the same consensus mechanism as Bitcoin. Peercoin was not created with scalability solution aim. Its aim was less energy consumption in mining step by utilizing PoS.

4.4 Attacks

Under this title, we will explain some past attacks and probable future attacks in Bitcoin and altcoins. Because of their different architecture they are vulnerable to different kinds of attack. But, generally they face similar attacks because of their peer-to-peer network architecture. We will discuss some precautions for these attacks and how their system can be robust to these attacks.

4.4.1 Bitcoin

In fiat currency, bank regulates money in the market. Bank or user is responsible for its security like theft or fraud. Bitcoin relies on a decentralized peer-to-peer network structure. So, it is open to some attacks like double spending, DoS (Denial of Service) attacks, transaction malleability.

- **Double Spending**

Double spending means attempting to spend same money for more than one thing. Attacker tries to spend same incoming transaction for different goods or exchanges. It is possible with so many ways like selfish mining, 51% attack, Finney attack etc. There are two ways to handle double spending attack. The first one is to detect fraud after it happened. The second one is to try preventing double spending with different protocols or regulations.

Bitcoin has verification system to tackle double spending problem. It marks transactions as spent or unspent according to their status. So, spent transactions cannot be spent again. But validation process can take some time because of Bitcoin protocol rules. In this process, two spending with same money can be attempted and can be successful because of network latency or attacker power on the network. Another case is similar, too. If attacker sends same money to merchant to buy something and its own another address at the same time, this is a possible successful

double spending. If merchant sends the good which attacker buys and transaction to the attacker's address is confirmed firstly before transaction sent to merchant, attacker has a successful double spending. Attacker gets own money back and gets the good without paying. But in this case, it should be guaranteed that the transaction to its own address must be confirmed first. For this aim, some tricks are needed like getting majority hash rate of the network or selfish mining etc. Thus, selected transactions get confirmation before other transactions created with double spending aim.

- **51% Attack**

Majority of network like greater than 50% of the network can consist of a group and can have hash rate more than rest of the network. Thus, they can create a private fork to blockchain and continue creating blocks until they get a longer chain than the rest of the network. When they publish the private chain, another fork becomes invalid. So, they can decide which block ultimately gets accepted as true. If there are two spent attempts, the majority can decide which one will be valid. So they can create double spends and they can validate the transaction which is addressed to their own address.

Formally, in mining, there are two parts; honest nodes and attacker nodes. Probability of finding a new block of honest node *p*, probability of finding a new block of attacker node *q=1-p*. z is the difference in heights between honest and attacker chains. If honest node finds a block, z increases with +1, attacker finds a block z decreases with -1.

$$z_{i+1} = \begin{cases} z_i + 1 \text{ with probability p} \\ z_i - 1 \text{ with probability q} \end{cases} \quad \textbf{(4.1)}$$

If q > p attacker controls majority of network and i goes to infinity, attacker succeeds double spending attack. This is called as >50% or 51% attack [76].

- **Possible Defenses Against to 51% Attack**

51% attack is theoretically possible. But in practice, after searching papers, forums, cryptocurrency websites to the best our knowledge, it has never happened till now except undetected ones. Even there is not any claim or rumor about 51% attack came true.

Till now, Ghash.io, one of the largest mining pool reached nearly 51% hashing power level three times [145]. But there is no evidence that they benefited from this attack. Sirer explains this situation with "Bitcoin runs over altruism" [146]. Because there was not any 51% attack during critical hash rate levels. He asserts there is altruism in Bitcoin by giving nearly 14 examples of some behaviors on the system. He explains that there is no attack during critic level because of altruistic miners and mining pools.

In Bitcoin system, when miners join larger pools, they gain more bitcoins. So, this incentive causes larger pools with time. Although, Sirer claimed Bitcoin runs over altruism. But Eyal and Sirer accept that a system cannot be secured depending on only altruism of users [147]. Thus, they proposed a new system to disincentivize large mining pools not to let miners create large mining pools to conduct 51% attack.

Their model to disincentivize large mining pools is Two Phase Proof of Work (2P-PoW) [147]. Greater than 25% hash powered mining pools are not good. Their 2P-PoW includes:

1. Calculation of (SHA256(SHA256(header))) is smaller than a difficulty parameter X as in Bitcoin.
2. Calculation of (SHA256(SIG(header, privkey))) is smaller than a difficulty parameter Y as the new step. (privkey is coinbase transaction's private key. SIG means header signed with privkey)

If the first phase difficulty is smaller and the second phase difficulty is higher than the first one, so many nodes find the first phase results and second phase comes as an important phase. In the second phase, there is a signature calculation of private key of coinbase transaction (means private key of node who has control on payment address) and pool operator cannot do the second phase own by own. Because it needs

a private key of node who finds the first phase solution. But key owner can find the second phase solution and take reward by itself. So, pool operator must choose participants more carefully. Also, if a pool wants to mine secretly, it must take all partial solutions. Making the first phase easier and making the second phase harder makes this operation very hard. Thus, 2P-PoW can be a good model to disincentivize large mining pools although it has some issues.

When Ghash.io reached the critical level like 50% hash rate, developers followed the process carefully in case of any attacks and Gavin Andresen from Bitcoin Foundation posted a warning that miners in Ghash.io should switch their mining pool like P2Pool or bitcoind urgently.

On the other hand, some researchers like Dave Hudson, chip architect with mining technology company PeerNova, and Andreas Antonopoulos assert that 51% attack is not possible only with 51% hash power. Because it is not logical to attack because of easily detected in blockchain when a pool have majority hash power. Double spend attacks cost them more than normal mining. Thus pool hash majority hash power does not prefer attacks [148].

Bastiaan proposed another prevention called two-phase proof-of-work (2P-PoW) rooted from the reseach by Eyal and Sirer [149] for 51% attack. They run their model on real world data and they assert that they get positive results.

- **Sybil Attack**

Attacker can create lots of peers to fill the network to manage some network attacks like refusing to relay blocks and transactions from everyone to a target peer for disconnecting the target peer from the network, sending only blocks which he creates for putting the target peer on a separate network, filtering some transactions for preventing the target hearing about other transactions. These manipulations can cause double spending attack.

- **Possible Defenses Against to Sybil Attack**

Douceur claims that any peer-to-peer system is vulnerable to sybil attacks without any central authority. But central authority is not a suitable network structure for Bitcoin idea. So, other ways must be experience to make Bitcoin network robust for sybil attacks. Rowaihy et al. proposed a solution to limit sybil attacks by creating an admission control system which checks joining nodes by using client puzzles [150]. Joining nodes try to solve puzzle from the leaves to the root. If joining nodes completes puzzle, they are given a cryptographic proof indicates that it can join the network. This is the first research that limits sybil attacks.

Another research by Bissias et al. proposed a sybil-resistant mixing for Bitcoin called Xim [151]. Xim is a two-part mixing protocol the first decentralized protocol detecting sybil attackers at the same time, detecting denial-of-service attackers. Xim increases cost o sybil attack for attacker linearly with total number of nodes(constant for honest nodes). Xim is more resistant to these attacks rather than Mixcoin, SharedCoin, Darkwallet, CoinShuffle. Also, Xim is fully compatible to Bitcoin.

- **Selfish Mining**

Eyal and Sirer defined *selfish mining* in their research which means a group of miner constructs a group and mine selfishly without publishing them to the whole network. Figure 4.6 shows state transition of selfish mining.

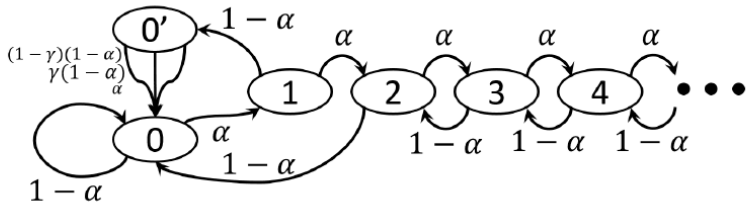

Figure 4.6 State transition of selfish mining [53].

They mine and publish their new blocks according to their algorithm as follows with description of its state machine [53, 143]:

State 0: If private chain is the same as public chain, attacker mines on private chain. If public chain network finds a block with probability $1-\alpha$, attacker resets private chain to public chain. If attacker finds a block with probability α, state goes state 1.

State 1: If private chain is 1 longer than public chain, attacker mines on private chain. If attacker finds a new block with probability α, private chain becomes 2 longer than public chain. If public chain network finds a block with probability $1-\alpha$, they have equal length and state goes 0'.

State 0': Attacker publishes its block and two alternative chains exist. If attacker finds a block, public chain is changed to private chain, state resets to state 0 and attacker wins 2 revenues. State resets state 0 in the rest probabilities, too.

State 2: If attacker finds a block with probability α, state goes to state 3 and attacker wins 1 revenue. If public chain network finds a block with probability $1-\alpha$, attacker publishes its private chain (2 blocks), and it is still 1 longer than public chain. Thus, private chain is changed to public chain. Attacker wins 2 revenues.

State n: For n>2, if attacker finds a block with probability α, it will be 1 longer than public chain and wins 1 revenue. If public chain network finds a block with probability $1-\alpha$, attacker's state goes back to previous state.

During attack, attacker has revenue and this revenue is a kind of incentive to join attacker's network, so other nodes join attacker's network to do profit maximizing. Thus, they asserted that Bitcoin is not incentive compatible and open to manipulate according to their research. This research shows that 1/3 threshold is upper bound for mining power of the network instead of 1/2 threshold wrong assumption. Protocol is not safe with larger than 1/3 threshold of any pool's hash rate.

- **Possible Defenses Against to Selfish Mining**

Eyal and Sirer proposed a solution to handle selfish mining that is backward compatible in Bitcoin protocol. When a miner gets two chains at the same length, it should relay both of them and choose chain to mine randomly. With randomly choosing chain at the same length, it will reduce selfish mining possibility.

- **Eclipse Attack**

Eclipse attack is about manipulation of peer-to-peer network. A node has 117 incoming and 8 outgoing TCP connections by default in Bitcoin network. But, this attack targets only the Bitcoin nodes that accept incoming connections because every node does not accept incoming connections. Eclipse attack is defined as getting control over a node's access to information in Bitcoin network. If attacker manipulates a node's connection, it can eclipse victim node to create a communication only with malicious nodes.

Eclipse attack is conducted as follows:

1. Attacker fill the node's peer tables with attacker IPs.
2. Node restarts and lost its current outgoing connections which are connected to the real Bitcoin network. Restarts occurs when there is update, DoS, power or network failure.
3. Victim node makes connection to only attacker IPs.
4. Attacker makes use of fresh timestamp property. Because victim node makes connections with fresher IPs in its IP table. Attacker's IPs are fresher because of attacker filling them newly. This is a vulnerability of Bitcoin. In fact, a node should choose its outgoing IPs randomly.

Attacker can launch a>50 attack with a smaller percentage hash rate. If network consists of three large mining nodes and two of them conduct 33% of total network hash rate and one, attacker has 34% hash rate, attacker partitions these two mining node, they cannot communicate each other. So, attacker becomes only one node who has majority mining power. It gets consensus blockchain with 34% hash rate without getting >50 hash rate of the network.

Heilman et al. simulated Eclipse attack with botnet of 4600 IPs, 2 IPs per group, 5 hours to fill the victim's IP table. They could eclipse a node with 86% probability for the worst case. The worst case represents that IP table is full with IPs with fresh timestamps [152].

- **Possible Defenses Against to Eclipse Attack**

To tackle Eclipse attack, Bitcoin protocol should choose IPs to connect randomly and Bitcoin should test node's connections to fill IP table faster if they are dead or not. Because filling IP table is slow and it has many dead node IPs.

- **Finney Attack**

Finney attack is a kind of double spending attack, if merchant or user accepts zero-confirmed transactions and attacker mines on his transactions even if attacker has smaller than 50% hash rate. Finney attack is a kind of Block Withholding Attack. Finney attack can be conducted as follows:

1. Attacker creates two transactions with same money, one of them is addressed to merchant, one of them is addressed to attacker's another address. But it does not broadcast the transaction sent itself.
2. It mines on the transaction created secretly and finds a block and it does not broadcast this block.
3. It broadcasts the transaction that is created for the merchant with the same money.
4. Merchant accepts payment and provides the service to the attacker.
5. Attacker publishes its secret block and its payment to merchant becomes valid. Because his payment to its own address gets confirmation firstly.
6. Finally, attacker gets its own money back and gets the service from the merchant without any payment.

- **Possible Defenses Against to Finney Attack**

In practice, merchants generally accept payments if payment has more than one confirmation. So, Finney attack is not prevalent attack for Bitcoin ecosystem. To not be exposed Finney attack, user must wait for more than one confirmations or use reliable third party like green addresses for faster transactions.

- **DoS Attack**

Attacker, individual or organization who has large computer network can flood so many data to a node that the node cannot process Bitcoin transactions. So, network cannot do its duty to confirm or relay transactions to others.

Mt.Gox was exposed to DDoS attack in 3 April 2013. This incident caused server lag and downtime and panic-selling. So, there was drop in Bitcoin price from $145 to $115 [153]. Mt.Gox declared that DDoS was conducted for monetary gain. It is said that attackers create DDoS and waits for dropping price of Bitcoin because of the panic-sell, then they stop the attack and they buy bitcoins at low price as much as they afford. They repeat it many times as it is seen in those days. Mt.Gox faced DDoS attacks so many times. In another incident, there was lag because of high volume rate in 10 April 2013 with dropping with a big Bitcoin price drop from $266 to $100 in nearly one hour [154]. DDoSers joined the incident with creating thousands of small value[20] orders and Mt.Gox suspended trade for a while nearly 10 hours because of upgrading servers [155]. After trade was started, Bitcoin price dropped to nearly $60 level because of downtime panic. As seen in Mt.Gox experience, it is clear how much influential effect DDoS has in Bitcoin network.

- **Possible Defenses Against to DoS Attack**

Bitcoin client has some precaution for DoS attack like below [157]:

1. A peer does not forward orphan transactions/blocks, double-spend transactions, the same block, transaction, nor process non-standard transactions.
2. Client bans misbehaving IP addresses for 24 hours as default.
3. Client keeps a DoS score of peers and disconnects it when score comes a limit.
4. Client limits the number of stored 10000 (by default) orphan transactions and 50000 (by default) stored signatures in the signature cache.
5. Client tries to catch all possible errors in transactions before the signature verifications take place, to avoid DoS attacks on CPU usage.

[20] From: http://i.imgur.com/cGjIS9x.png Retrieved 14 March 2016

6. Client considers non-standard signature scripts that contain operations that are not PUSHs and non-standard signature scripts with size greater than 500 bytes.

There are also some protocol rules:

1. Protocol restricts the block size to 1 megabyte, the maximum number of signature checks a transaction input may request.
2. Protocol limits the size of each script up to 10000 bytes, the size of each value pushed while evaluating a script up to 520 bytes, the number of key arguments OP_CHECKMULTISIG can use up to 20 keys, the number of the stack elements that can be stored simultaneously, the number of signature checks a block may request up to 20000 checks.

- **Wallet Theft Attack**

Wallet stores public-private key pair. If private key is stolen, coins of user can be stolen by attacker. So, wallets must be secure to protect private key. Wallets are secured with a password to prevent access of others. But these passwords can be weak because of memorizing problem. So, attacker can hack wallet easily with a small amount of power. Researchers attacked brain wallets as known electronic wallets and got $103, 000 from 884 users' wallet [158]. They created a huge words list consists of 300 billion passwords. They created this words list from English word lists, urban dictionary, slang dictionary, English Wikipedia, phrases, lyrics etc. After nearly 4 years evaluation from September 2011 to August 2015, they found 845 different passwords from 884 user brain wallets got 1806 BTC. It is obvious that wallet security is vital for Bitcoin ecosystem.

- **Possible Defenses Against to Wallet Theft Attack**

Bitcoin Project Team declared some advices how to secure Bitcoin wallet [159]:

1. Strong passwords for brain wallets.
2. Two-factor authentication to access to wallet.
3. Store less money in hot wallet.
4. Backup wallet in case of any computer failure and encrypt back up.

5. Locate backup in many locations in case of single-point-of-failure.
6. Keep software up to date.
7. Use multi-signature for transactions.

Also, Pham and Lee proposed an anomaly detection method by using unsupervised learning methods. They tried to detect anomalous users and transactions and they can detect two types of theft and a case of loss [160].

4.4.2 PeerCoin

Peercoin uses PoW and PoS at the same time. So, it has different protocol rules against attacks like checkpoint rule.

- **Checkpoint**

To prevent any possible changes in blockchain history and lower double spend attack, checkpoint are utilized in Peercoin. Firstly, they attempted to design a decentralized checkpointing but it was not a good design to secure system easily. So, creators of Peercoin designed centrally broadcasted checkpointing against DoS attack and 51% attack. Central checkpointing checks past transactions older than one month.

- **51% Attacks**

For 51% attack, 51% PoS and 51% PoW power is not necessary at the same time. Peercoin security depends on only PoS although its mining system depends on hybrid system with PoS and PoW. Also attacker does not need to have >50 coins of Peercoin system. It needs to have significant proportion of minting coins in the network. If miner decreases, monopoly increases and 51% attack possibility for PoS gets higher. A user can hold money at most 90 days in its account to create high coin age. After 90 days, user's coin age is stable until it adds new coins to his address.

Except energy consumption, price and market capitalization, Peercoin has similar structure to Bitcoin structure in terms of privacy, scalability, transaction malleability, fungibility.

4.4.3 Zcash, Ripple, Dash

There are not any reported or proposed attacks for Zcash. But as all peer-to-peer networks, Zcash will be vulnerable to DDoS attack. Also, there is an unclear concern about Zcash that 51% attack can be mounted because of latency caused by forwarding and verifying all blocks at each hop. Bitcoin cashes transaction verifications so that there is not any significant latency. Also, simulation results show that Zcash has very negligible latency even though each hop verification process [5].

Ripple charges transaction fees dynamically so that transaction spams do not fill the network. Also, each Ripple account needs 20 Ripple (XRP) reserve to make harder any new spam node joining the network. 51% attacks means in Ripple that a group of malicious group gets control over majority validators; consensus fails and malicious group can validate transactions first whatever they want. But users selects validators specifically them who do not collude with malicious third party. So, 51% attack is harder to conduct in Ripple network. Furthermore, if validators refuse to come a consensus, other validators are informed about this and network stops the service because it cannot be detectable who part is right about refusing consensus [161]. Ripple has preventing mechanism for DDoS attack. It increases transaction fees if number of transaction increases so that attacker cannot create more transactions at high cost. This limit is defined by protocol dynamically.

Dash advices smaller pool sizes against to 51% attacks. Also, Dash proposes a transaction to users sent to them when they join the network. This is a precaution for DoS attack. While CoinJoin is vulnerable to tracing attack, Darksend is not as much as vulnerable to some tracing operations to track user activity. Because Darksend uses mixing rounds and 3 clients in one mixing.

4.5 Future Perspective and Problems

Finally, at the end of the fourth chapter, we will present future perspectives and problems for Bitcoin and other Bitcoin. With time, new altcoins and new attacks for both Bitcoin and altcoins are studied and discussed. But it is clear that cryptocurrency technology will improve and be prevalent in the market and finally replace with fiat currency in the future world.

4.5.1 Bitcoin

Bitcoin is a promising technology for currency system because of its decentralized structure. But some concerns have been emerging by researchers and community. With increasing transaction volume, Bitcoin network and peers reorganize their structure to handle great amount of transaction flow. As first attempt, block size is reconsidered allowing miners mining of more transactions in one block. With these solutions, blockchain size and network bandwidth problems become current issues. For full nodes, handling terabytes of data is very hard work and this issue needs an extensive solution. Because if Bitcoin is a cryptocurrency as an alternative to fiat currency, it must handle great amount of transactions like PayPal 115 transaction per second while Bitcoin has ability to handle 7 tps [75].

Also, with halving mining reward, in the future, transaction fee will increase to supply an incentive to miners to get confirmations. This will result new Bitcoin system that is not desired as a Bitcoin idea. Bitcoin aims money transfer and shopping with low transaction fees.

In the future, Bitcoin network will be exposed to transaction spams like nowadays but with a stronger attack. Bitcoin network is vulnerable to spam attacks as we saw how badly systems are affected like the Mt.Gox incident. So, last experiences give idea about attacks to Bitcoin network in the future.

In terms of energy consumption, Bitcoin will use more energy to mine new blocks because of increasing difficulty. This will cause lots of debates for Bitcoin future in terms of energy consumption [162]. At the time of writing, hash rate, measuring unit of the processing power of the Bitcoin network, is 1,365,664,121 GH/s [21]. It means that Bitcoin network can calculate $1{,}365{,}664{,}121 \cdot 10^9$ hashes (nearly 1367 petahash) per second. If we assume that 10 watts is per Gh/s, it will be 13656641210 watt

[21] For more information: https://blockchain.info/charts/hash-rate 29.03.2016

means 13, 656. 64121 megawatt per second, 327, 759. 36 gigawatt-hours per day, 119,632,166 gigawatt-hours per year, 119, 632.166 terawatt-hours per year. This number corresponds to 25 times of China electricity consumption in 2014. China had 4833 TWh electricity consumption in 2014 [163].

Volatility is also another problem for Bitcoin. Because its market price changes very fast in a short time. So, many people lost money by investing money on Bitcoin. But sometimes people can make a large amount of money like between October 2013 and November 2013 rising from $125 to $1,124 in Mt.Gox. Then it fell from $1,124 to $415 gradually till now. It is obvious that Bitcoin does not have volatility[22] like fiat currencies. So, users give up using Bitcoin for trading for this reason. Although it has volatile price history, it is believed that it is not a fatal problem for Bitcoin [154].

4.5.2 Altcoins

Users prefer different altcoin for different reasons like privacy, fungibility, faster confirmation time, multi-functionality, durability to attacks, less volatility, sustainability, less energy consumption and these properties will shape of future of Bitcoin and altcoins.

For privacy, Zcash is the most powerful rival for altcoins and Bitcoins. Because most of users prefer untraceable payment history. Because they do not want to share their trade interest even if they do not use altcoins in black markets or money laundering. This preference depends of user's sensitivity about life way or trading privacy due to cover up some illegalities.

For easy fungilibility and less transaction fees, Ripple has great opportunity to have place in altcoins in the future. Because it allows exchanges between many currency types including fiat currency, gold, silver. Also, it has small amount of transaction fees for exchanges rather than PayPal, Visa. It seems like that users who trades inter currencies prefer Ripple for the future.

---

[22] For more information: https://blockchain.info/en/charts/market-price 30.03.2016

For energy efficiency, proof-of-stake based cryptocurrencies like Peercoin, Nxt is preferred by users. Because users cannot have a great amount of power to find a suitable hash like used in hashcash proof-of-work based cryptocurrencies. So, users as miners they prefer using less power needing systems. Also, for users who have energy consumption concerns for saving the earth they prefer PoS based. Some community asserts that Bitcoin mining is an environmental disaster [164].

4.6 Discussion of Comparison Bitcoin and Altcoins

Bitcoin leads as the first successful crytocurrency in the world. Its successors are altcoins. Altcoins have been developing for different deficiency in Bitcoin or with innovative ideas.

Bitcoin has not privacy. So this brings new innovative crytocurrency ideas like Anoncoin with extensions for I2P or Tor anonymous networks, Bytecoin with ring signature technology, Dash with mixing services, Zcash with cryptographic protocols like zero-knowledge proofs. But these altcoins do not have market capitalizations as large as Bitcoin and price as high as Bitcoin price. Bitcoin has longer history than altcoins. So, it has more investors and users as normal. But it is foreseen that Zcash will find a high majority of users because of its fully private transaction property. Although, some communities have criticisms that untraceable transactions cause money laundering, tax evasion and higher trading in black markets with illegal drugs, weapons, human trafficking, stolen credit cards, fake driver licenses, child pornography. These criticisms are valid for all anonymous altcoins.

The second must improvement for Bitcoin is scalability. As its first years, scalability was not a big deal because of its small size blockchain and network size. But with time scalability debates raised. Bitcoin community and developers started to improve different proposals like BIP family and innovative blockchain ideas like Bitcoin-NG. But it has some security concerns. Also for scalability, some blockchain pruning developments have been studied by researchers. There is a tradeoff between security and scalability in the big picture. As we explained in scalability chapter, some scalability improvement proposals have pitfalls in terms of security. Developers and

community of Bitcoin settled on a consensus about scalability and we will experience in future 1-2 years how it will work for security and scalability.

CHAPTER 5

DIFFERENT APPLICATIONS OF BLOCKCHAIN IDEA

In this chapter, we will explain new innovative altcoins, which benefit from blockchain idea. They are not only currency technology but also they provide different applications like independent domain name system, notary system and programmable blockchain.

5.1 Namecoin

Namecoin is the first fork of Bitcoin software. It uses PoW and is limited to 21 millions of namecoin and reward is halved in every 4 years by starting from 50 NMC. Its symbol is ℕ or NMC. Namecoin aims serving as a domain name service (DNS) independent from ICANN (Internet Corporation for Assigned Names and Numbers) with .bit top-level domain like .com domain. Identity system, messaging system, notary systems, alias systems are other potential usage of Namecoin.

About domain name services, there is a restriction about any naming system called Zooko's Triangle.

5.1.1 Zooko's Triangle

Zooko Wilcox O'Hearn proposed a model that explains 3 properties of a naming system and claims only two out of three properties can be owned by a naming system.

1. Secure: A domain name maps unique entity and nobody can pretend as if owner of its name.
2. Decentralized: A decentralized system decides on meaning of a name.
3. Human-meaningful: Names are enough short for human memorizing.

Three choices according to Zooko's triangle:

1. Secure and Decentralized: This type does not have human-meaningful property. For example, .onion addresses like ueysbjsuete73hdtehrt.onion. They are not human-meaningful.
2. Secure and human meaningful: This type does not have decentralized property. For example, DNSSec (Domain Name System Security Extensions)
3. Decentralized and Human-meaningful: This type does not have security property. They relies on trusting third parties. For example, I2P (Invisible Internet Project) like example.i2p

Swartz disproved Zooko's Triangle model by saying that it is possible to create domain name service contains all three properties [165]. Naming system based on Bitcoin makes his disprove possible. Because it fulfills three properties at the same time. Namecoin is secure, decentralized and human-meaningful.

Namecoin records contain key, value pairs. Key represents a path with DNS namespace *d* like *d/example* with name *example* corresponds to *example.bit* address, value shows data to attach this address by sending special transactions and storing in blockchain. In Namecoin, keys are secure, decentralized and human-meaningful.

5.1.2 Domain Namespace

For registering a domain:

With Namecoin RPC commands `name_new`, `name_firstupdate` users register own domain names [166]. Commands are created in Namecoin client `namecoind`. For example, `name_new` command reserves desired domain name by sending a transaction with hash of domain name with salt and transaction id. This is a precaution for the domain name for preventing others stealing this domain name. After 12 block confirmations, `name_firstupdate` is accepted. Because while a `name_new` transaction is waiting for a confirmation, with the same domain named `name_new` transaction can get confirmation faster than the first one. Thus, 12 confirmations duration is waited for acceptance of `name_firstupdate`. If a domain name collision occurs, older `name_new` transaction wins. When `name_firstupdate` transaction is added to the blockchain, this domain name is

registered for that user and domain name becomes valid. A special coin is tied to this key, value pair with 0.01 NMC. User's wallet stores this special coin. Registered domain expires after 36,000 blocks nearly 250 days after its register or last domain update. These special coins cannot be used in normal payments because Namecoin core prevents it. With `name_update` command, domain values can be updated by its owner.

For resolving domain:

Top-level domain .bit used for domain names of users is stored in Namecoin blockchain. Regular DNS servers cannot resolve .bit domains. They return errors to indicate that there is not top-level domain called .bit. So, for .bit domains, a new DNS server is needed to resolve addresses. NMControl is kind of domain lookup software. It runs like a local DNS server. It needs a full copy of Namecoin blockchain and a full synchronization with it. It has namecoind client to copy of the blockchain. NMControl takes resolving request (domain names) and checks from the blockchain and responds the corresponding IP address. There is similar lookup software like NMControl for resolving .bit domains.

5.1.3 Differences Between Bitcoin and Namecoin

Namecoin is a fork of Bitcoin code but its aim is totally different from Bitcoin. Bitcoin is a currency for trade. But Namecoin uses blockchain as domain name server storage and allows regular payments. With confirmed transactions, key-value pairs are stored in the blockchain for later queries. Namecoin conducts a decentralized and independent from ICANN domain name system. If domain name is controlled by a centralized third party, it is easy to be banned by governments or hacked by attackers. Thus, Namecoin created decentralized domain name system without any hierarchy and all records are open to public by using the block chain. While DNS registrars are based on profit, Namecoin does not follow monetary profit. Users only pay miners for transaction like name_new, name_firstupdate, name_update. This is kind of incentive to miners to validate user's transactions which are for registering or updating a domain name.

Namecoin has some problems like privacy, transaction malleability and energy consumption similar to Bitcoin.

Namecoin is exposed to some attacks like 51% attack allowing double spending and name theft. Any attacker has 51% hash power can steal newly registered domain names or prevent user updating domain names by not validating user's transactions. Because attacker with 51% hash power can validate own transactions faster than other users'.

In terms of scalability, for now there is not a significant scalability problem with 4.18 GB blockchain size or problematic network infrastructure [23]. It has compression mechanism for blockchain to make a smaller-sized such 250MB.

Namecoin is a very innovative utilization of blockchain idea by serving a new decentralized domain name system.

5.2 Ethereum

Ethereum was firstly proposed by Vitalik Buterin in 2013. Ethereum project was released on 30 July 2015 by Gavin Wood and Vitalik Buterin [9]. Ethereum is a cryptocurrency with smart contract idea relies on blockchain. Smart contract is a computer protocol that it serves as a decentralized virtual machine that runs peer-to-peer smart contracts using cryptographic algorithms called cryptographic asset *ether, ETH*. Ether is used in trading and exchanges like bitcoins. For ether mining, Ethereum uses proof-of-work and switching to proof-of-stake is planned in near future because of mining cost, scalability and decreasing censorship. Block rewards are constant so number of ether will be unlimited in the system.

Ethereum allows people create cryptocurrency, voting systems, bank systems, auction systems, websites, social networks, games, blockchain explorer software etc.

Ethereum has three types of applications. The first one is financial applications like new altcoins, financial derivatives, saving wallets and wills etc. The second one is

[23] For more information: https://bitinfocharts.com/namecoin/

semi-financial applications. There is money in these applications but mostly non-monetary applications like self-enforcing bounties to solve computational problems. The third one is not financial. They are used for decentralized governance operations like online voting.

5.2.1 Ethereum Accounts

In Ethereum, each state consists of accounts that have 20-byte address. Value and information transfer between accounts are state transitions. An account consists of current ether balance, contract code (if present), storage and a nonce that ensures a transaction occurs once.

Ether is defined as a cryptographic fuel of Ethereum system and is used as transaction fee by users.

It has two kinds of account:

1. **Externally Owned Accounts (EOA):** Private keys controls EOA. Users control private keys. User creates and signs transactions and send from own EOA.
2. **Contract Accounts (CA):** Contract code controls CAs. Only EOA activates CA. Internal code rules CAs. A contract account receives a message and its code activates it and it becomes a readable and writeable by internal storage. Also, it sends messages and creates contracts.

Contracts are kind of control mechanism that is not need to be fulfilled or compiled with. They stay in Ethereum execution environment. They are activated by message or transactions. They control over ether balance and key-value storage and have ability to track some variables in contract [167].

5.2.2 Transactions

Transaction is a signed data that stores message and sent from an externally owned account. Transaction includes the receiver of the message, signature to identify the sender, amount of ether to transfer from sender to receiver, optional data field, a startgas value to show the maximum number of computational steps which

transaction execution is allowed, a gasprice to show fee for sender's payment for per computational step. Sender, receiver and amount of coin are the same as other coins. Ethereum uses three additional data. Gasprice and startgas are precaution for denial-of-service attacks. Attacker cannot flood the network with many transactions. Because it needs many ethers to conduct the attack. Gas is the unit of computation. Each computation step costs generally 1 gas. Cost depends on the kind of computation. If computation of the step is computationally expensive, it costs more than one gas. Also there is 5 gas fee for every byte of transaction. Also this fee is a kind of anti-DoS functionality [168].

5.2.3 Messages

Contracts can send messages to other contracts. A message contains sender, receiver, and amount of ether to transfer the message, an optional data field and a startgas value. Message is a kind of transaction created by only a contract with its CALL opcode.

5.2.4 Ethereum State Transition Function

Ethereum state transition function is defined as APPLY(S, TX) → S', S represents State, S' represents State' and TX represents transaction. Account state changes to state' after transaction operation. In the state transition example, owner of account sends 10 ether to Charlie who is owner of bb75a980 address. Figure 5.1 shows Ethereum state transition example.

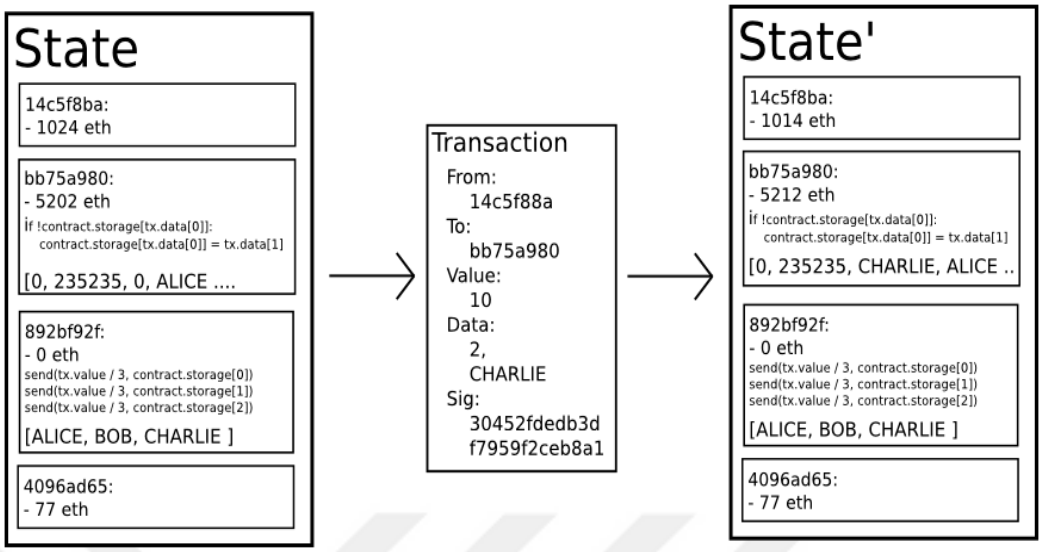

Figure 5.1 State transition example [168].

Ethereum state transition function is like below [168]:

1. Transaction form, validation of signature, nonce are checked. If at least one returns false, it returns error.
2. Transaction fee is calculated as startgas · gasprice. Sending address is determined from the signature. Fee is taken from sender's account. Sender nonce is incremented. If sender account balance is less than the fee, it returns error.
3. Gas = Startgas is initialized. Gas per byte is calculated.
4. Transaction value is transferred from sender to receiver. If the receiving account is a contract, run the contract's code until gas ends.
5. If sender does not have enough funds to send or gas ends, all state changes are reverted and transaction fee is added to miner's balance.
6. Else, fees for all remaining gas is refunded to the sender, and the fees paid for gas consumed to the miner is sent.

There is an example about this process. It is assumed that 10 ether is sent to Charlie with 1000 gas 0. 001 gasprice and 64 byte data with 0-31 representing number 2, 32-63 for Charlie name. It work like below:

1. Transaction is checked.
2. 1000 · 0. 001= 1 ether Subtract from sender balance.
3. Initialize gas=1000 Transaction size is assumed as 160 byte. Fee is per byte is 5. 5 · 160=800 1000-800=200 gas is left.
4. 10 ether is subtracted from sender balance and added to receiver balance.
5. Code is run. Storage index set to Charlie if it is not used. Code running is assumed as 150 gas cost. 200-150=50 gas is left.
6. 50*0.001=0.05 ether is sent to back to sender.

5.2.5 Ethereum Virtual Machine

It serves as a platform of many blockchain applications at the same time. It can execute any complexity code. So, it is "Turing complete".

Ethereum includes peer-to-peer network because of its decentralized architecture. Ethereum blockchain is stored in every full node. Each nodes runs EVM. Contract code is written in the low-level EVM code. EVM serves as a platform that developers create applications and run on it. So, it is called "world computer". Due to its decentralization, it is fault tolerance and has zero downtime by saving data unchangeable. Thus, it is called programmable blockchain. Programmers decide what they implement with it with any programming language. But some applications are more suitable Ethereum like peer-to-peer marketplaces.

5.2.6 Code Execution

Ethereum contract code is written with a low-level stack based bytecode language. It is called Ethereum virtual machine code. Stack, memory and key-value storage are used in EVM code. A whole computational state is defined as a tuple *(block_state, transaction, message, code, memory, stack, pc, gas)* and *block_state* is a global state that contains all accounts, balances and storages. Each instruction makes changes in the tuple.

5.2.7 How Does Ethereum Works?

While bitcoin blockchain consists of transactions, Ethereum blockchain contains accounts. Ethereum tracks account activity.

Smart contracts are defined as a code in a contract account. They are implemented with any language and compiled into bytecode for EMV before assigning to the blockchain. All contracts are stored in the blockchain publicly so that anyone can verify them. But EMV codes are difficult to interpret. So, contract is compiled and compared with corresponding EMV code. So, smart contracts are transparent.

Ethereum uses *gas* to limit smart contracts in case of a long time to run contracts. Because contract computation on EMV is expensive. A certain amount of ether is determined per computation.

Users pay transaction fees to miners like Bitcoin. Miners verify transactions in blocks with PoW with Keccak-256. It is planned to switch to PoS in the near future.

5.2.8 Ethereum Blockchain and Mining

Ethereum blockchain is similar to Bitcoin blockchain. But it has some differences because of its architecture. Ethereum blockchain includes all transaction lists and the recent state. Figure 5.2 represents Ethereum blockchain.

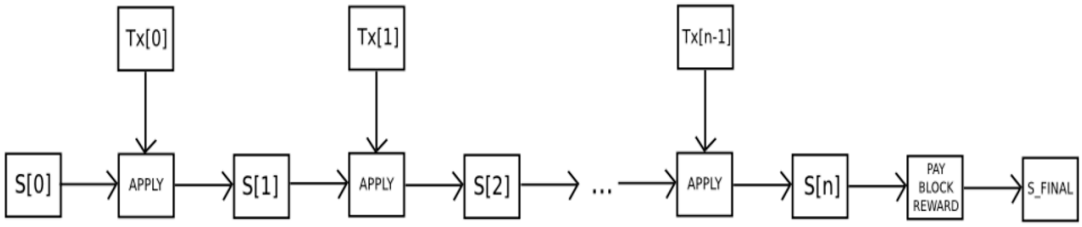

Figure 5.2 Ethereum blockchain [168].

Ethereum block validation is like below [168]:

1. Previous block reference is checked if it exists and is valid.

2. Timestamp is checked if it is greater than the last block timestamp and 15 minutes difference.
3. Block number, difficulty, transaction root, uncle root and gas limit are checked if they are valid.
4. Proof-of-work of the block is checked if it is valid.
5. The state at the end of previous block is called as *s[0]*.
6. *TX* is the transaction list with *n* transactions. Apply for all n from 0 to n-1 *S[i+1]= APPLY(S[i],TX[i])*. If any error is returned and used gas is greater than the gaslimit, it returns error.
7. S[n] is the final state of the block. Block reward is paid to the miner.
8. Merkle root of the final state and final state in the block header is checked if they are equal. If they are equal, block is valid. Else block is not valid and is not added to blockchain.

5.2.9 Differences Between Bitcoin And Ethereum

While Bitcoin is currency just used for trading and exchanges, Ethereum is a platform allows developers create different applications on it. It is not a dynamic blockchain; it is programmable blockchain with smart contracts in a transparent process. Bitcoin is the Internet money; Ethereum is called as world computer because of its different blockchain innovative idea.

In terms of privacy, Ethereum is similar to Bitcoin structure with traceable transaction on the blockchain. So, Ethereum users have privacy concerns, too.

In terms of scalability, it seems like that Ethereum will face blockchain and network scalability problem. Scalability is eternal concern for most of altcoins.

In terms of transaction malleability, Ethereum has transaction malleability problem. But it did not face a serious problem unlike Bitcoin. Vitalik Buterin declared that even if Ethereum is exposed a transaction malleability problem, its consequences are not big unlike Bitcoin. Because only transaction reference accounts are affected not other transactions [184]. But Vitalik Buterin claims that they will have a soft fork and after this patch they will have not any transaction malleability problem in Ethereum [184].

In terms of energy consumption, Ethereum uses PoW and will face similar problems like Bitcoin. But when Ethereum switches to PoS system called Casper, it is aimed that it will have much lower energy consumption [168].

In terms of attacks, Ethereum is vulnerable to %51 attack, long-range attacks [169, 170]. Also, one of the applications of Ethereum, ShellingCoin, is vulnerable to P + Epsilon attack [171].

5.3 Colored Coin

Colored coin is a different application of blockchain idea. It aims to exchange different kinds of assets by using blockchain. Any kind of real world asset like car, house can be exchanged between users on blockchain securely. Scripting language of Bitcoin allows storing some metadata on blocks so that asset is this asset transaction data can be written in blocks on the blockchain. Colored coin wallet creates this transaction and sends to the network. Blockchain provides immutability, non-counterfeitability, ease of transfer, robustness and transparency. User can publish shares/tokens by using colored coins for trading, voting. Colored coin can be used as a smart property like representing a car for rent, a coupon, community money for a local community, digital collection like song, software and access token like museum card or subway card. Also, colored coin can be used as a lock to open a door of house or car in case of door identifies that specific coin like key.

The basic idea is to stamp a bitcoin with a color similar to stamping data in a currency. It becomes still a valid bitcoin and it also contains some data. This stamping process is called as issuing. This data implies an output that has a specific color. Color represents some bit strings to illustrate colored coin idea. If this bit string is the same as other coin's color, it means they have same color. Same color coins can be merged in a big same color coin.

5.3.1 OpenAssets

OpenAssets is implemented in Bitcoin with colored coin idea and compatible with Bitcoin. OpenAssets is integrated into NASDAQ global electronic marketplace in 2015 to transfer assets in collaboration with blockchain. Pay-to-Script-Hash address

is utilized to issue assets. After choosing a P2SH address, coin can be issued as a colored coin through P2SH address. Corresponding address should be public so that corresponding address to color is known. Also, a coin can be issued more than one color issuing address that means a coin can have more than one color.

Incoming colored coin is divided into different values by using data in the output called special marker output.

Disadvantage of OpenAssets is to be obligated to create an unspendable output (coin) for trading each colored coin. This unspendable output is created by OP_RETURN Script language. This opcode is used to mark a transaction output as invalid means unspendable. This is a kind of storing data method in blockchain. Also, miners do not verify colored coins, but verify underlying bitcoins. User checks all transaction history of colored coin to check validity of the colored coin or trusts a third party who checks validity of colored coin. SPV clients cannot be used for colored coin. So, limited memory devices could not create transactions with colored coins. But CoinSpark developed a SPV desktop wallet and Colu developed an application for mobiles to send digital assets with colored coin scheme.

5.3.2 Coloring Bitcoin Transactions

There are two opcodes to use in coloring transactions called OP_RETURN and Multisignature address called multisig. OP_RETURN is used to embed data in blockchain. It allows storing a small data on the blockchain. Bitcoin network accepts at most one OP_RETURN. Colored coin protocol uses 1 out of 2 or 1 out of 3 multisignature addresses to avoid extra complexity. Up to 80 bytes Asset manipulation data is stored after OP_RETURN opcode.

There are two types of asset manipulating transactions called issuance and transfer. Data called metadata is stored by using torrents to provide decentralized sharing and storing. So, metadata is not stored on blockchain directly. In issuance transaction, new asset is created firstly. In transfer transaction, assets are only transferred [172].

5.3.3 How does Colored OpenAssets work?

Colored coin allows storing user's good in a coin and exchanging its ownership by using a colored coin transaction like Bitcoin transaction. For these operations, there are many wallets like Coinprism, Colorcore, Colu, CoinSpark, ChromaWallet.

OpenAssets uses Open Assets Protocol to store and transfer assets on the blockchain. Firstly, issuer issues colored coin under a promise that user can redeem later according to terms that he defined in issuing step.

Outputs that use Open Assets Protocol have two new properties called asset id to identify colored coin uniquely, asset quantity to represent quantity of stored units of this assets on the output. Uncolored outputs do not have asset id and asset quantity properties.

Asset id is calculated like RIPEMD160(SHA256(script))). A colored coin can be reissued by the same private key. If two colored coins have same asset id, they can be mixed together.

To generate an asset id [173]:

1. Issuer generates a private key.
2. Issuer calculates address with this private key. This guarantees that nobody else can issue assets with this asset id.
3. Issuer creates Pay-to-PubKey-Hash script for corresponding address.
4. Issuer hashes this script.
5. Hash is converted base 58 with checksum. This is the asset id.

Open Assets Protocol has output called marker output. Thanks to marker output, clients understand that this transaction is an open assets transaction. Open Assets transactions allow issuing new assets and transferring ownership of asset. If client does not recognize an Open Assets transaction, this transaction contains uncolored outputs.

Marker output starts with OP_RETURN opcode and contains other opcodes. But in these opcodes PUSHDATA opcode must place storing Open Asset marker payload to parse.

With a basic scenario, we will explain using of colored coin in figure 5.3. Lets say Alice has 100 Facebook shares. Bob wants to buy 25% of Facebook shares of Alice and Carlisle wants to buy 75% of Facebook shares. A Facebook share price costs 100 USD (0. 22 BTC). Carlisle and Bob trusts Alice. Today, a bitcoin costs 450 USD. Bob will pay $25 \cdot 100 + 0.25 \cdot 450$(for 0.25 BTC) equals 2612. 5 USD (or 5. 805 BTC) to Alice. Carlisle will pay $75 \cdot 100$+0. $75 \cdot 450$ (for 0.7 5 BTC) equals 7837. 5 USD (or 17. 416 BTC) to Alice.

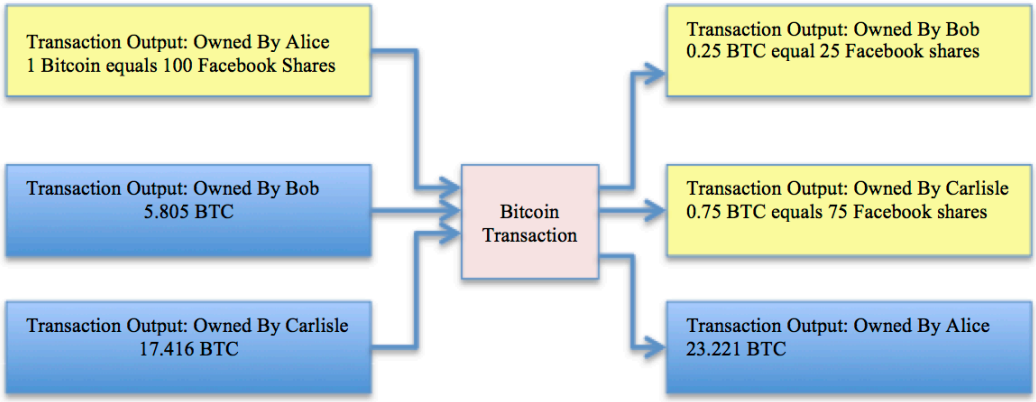

Figure 5.3 Colored coin transaction from Alice to Bob and Charlie.

5.3.4 Asset Verification

Colored coin supports asset verification by linking a social account of user like Twitter, Facebook, Github account. User needs to add its twitter handle of its twitter account like @your_twitter_handle under the asset metadata verifications key. After asset is issued, takes the asset id and tweets asset id as a hashtag. With Facebook, User creates a page with his own Facebook account and takes page id and adds it to the asset metadata verifications key. After asset is issued, user publishes its asset id

with a hashtag on his Facebook page. Another way for verification, to have a website and a SSL certificate. User must host the metadata file on his server including in a json file with asset id, contract url, name, issuer, description etc. When user issues his coins, he must specify the metadata url. Besides this, user must trust issuer. Because colored coin is trust based scheme when accepting an issued coin.

Advantages of Colored Coin

1. It allows to represent any kind of asset with colored coin and transfer them easily.
2. It allows storing these assets without any third part. Their security is provided by Bitcoin scripting language.
3. Their transfer is easy to conduct without any central authorization.
4. Any kind of assets named after colored coin can be exchanged with colored coins or uncolored coins easily in a faster and efficient atomic transaction.
5. All transactions can be verified by using blockchain that cannot be manipulated.
6. Ownership of colored coin can be made anonymous so that user has private transactions under some precautions.
7. Colored coin benefits from Bitcoin infrastructure, which is very prevalent and widely used system.

5.3.5 Differences Between Bitcoin And Colored Coin

While Bitcoin only represents funds by using blockchain, colored coin utilizes each coin as different kinds of assets like house, car, bonds, stocks. Bitcoin is fungible with other kind of currencies via different exchange third parties. But, colored coin provides exchange ownership of any asset without need of third party easily and at a very low fee. Colored coin cannot be spendable like bitcoin. Their ownership is exchanged with transactions. But receiver can pay with any currency using exchange third parties.

CHAPTER 6

DISCUSSION AND CONCLUSION

Bitcoin is the leader and by far the most widely used cryptocurrency. So, many researches and development have been studied about Bitcoin and altcoins. In this thesis, we discussed technical background of Bitcoin altcoins like Zcash, Ripple, Peercoin and Dash. Also, we compared Bitcoin and these altcoins in terms of different deficiencies of Bitcoin.

In privacy, Bitcoin has some problems like traceable transactions and disclosing users identity. To handle this problem, Zcash has been developed. Zcash promises private transactions by using ZK-SNARKS cryptographic schemes.

In transaction malleability, Bitcoin faced a major attack. It was an implementation of Bitcoin transactions of exchange company. But, it is seemed as if a problem is in Bitcoin's source code problem. But after this incident, all exchange clients were patched. Also altcoins rooted from Bitcoin patched their client software against to transaction malleability attack. After that, Zcash proposed transaction non-malleability scheme for this kind of vulnerability. So, Zcash is resistant to transaction malleability attacks in the future.

In scalability, it is very clear that Bitcoin will face a big scalability problem in the future. So, Ripple developed a new payment network by using off-chain transactions by increasing number of transactions per second. But scalability will be a common problem for Bitcoin and all altcoins. There is not any found accurate method to handle scalability problem in decentralized systems. There are some proposals for scalability problem that solves some particular problems in scalability.

In terms of attacks, Bitcoin and generally altcoins have vulnerabilities to attack successfully. Because cryptocurrencies have new different infrastructures and unknown properties. With time, new explorations will be found and will be taken

new precautions for new vulnerabilities. Cryptocurrency is a new interest area for users and developers. So, it cannot be predicted which vulnerability will be detected in Bitcoin and altcoins.

We presented an extensive Bitcoin and altcoin research based on academic researches and online resources. We compared Bitcoin and some altcoins by using different perspectives. It showed that many new researches for cryptocurrency world have been studied every day. Along with new development, it showed there is not any flawless cryptocurrency yet. Because there are many aspects to examine for full-fledged cryptocurrency concept. In spite of these many defects of Bitcoin and altcoins, they are highly popular in the market and they got significant attention of users for their different properties. But still any altcoin could not get attention and market share as much as Bitcoin.

There are many aspects to improve like privacy, scalability, security, energy consumption, new blockchain ideas. After Bitcoin, nearly 2700 different alternative coins have been invented for different usage areas and aims [10]. Altcoins have become popular topic in cryptocurrency space. New altcoins with different mining algorithms like Litecoin, Primecoin have been developed. A new altcoin called Peercoin replaced mining with a lightweight method and Ripple developed a new cryptocurrency network to exchange all kinds of coins including fiat currencies in an efficient way. Dash was created for untraceable transactions by using mixing services. After that, for privacy concern Zcash was developed by a group of crypto experts by hiding sender, receiver and amount of transaction. With Bitcoin, blockchain become very popular technology to implement very different applications on it. Namecoin, Colored Coin, Ethereum benefits from blockchain features to implement innovative applications. It is believed that there will not be any payment system without blockchain technology for secure transactions. Blockchain is not only promising for payment technologies but also it will be an essential part of smart contracts, notary operations. Since 2009, lots of altcoins came to the market and a new one is being developed every week. Bitcoin opened a new era for the future of money and new breakthroughs.

In this thesis, we planned to develop some improvements for Bitcoin architecture for some problems like privacy, scalability and vulnerabilities to attacks. But Bitcoin network is too large to test Bitcoin architectural improvements. Bitcoin fixes are hard to deploy to test its performance. Although it has a testnet to test Bitcoin improvements, it needs to have a large number of users to see results of the test. For Bitcoin scalability, we considered to improve sidechains, treechains and micropayment methods in this thesis. But Bitcoin is a big network and it is hard to adopt an improvement. So, we will leave these improvements as a future project as open problems or to large communities who have large network environments and hardware. Also, for privacy some zero-knowledge protocol fixes can be possible for Bitcoin. For small third parties like us, new development testing is not easy to conduct for now. So, in this thesis we could not test new improvement to Bitcoin because of hardware and network constraints. For now, we leave these problems as open problems for future researchers.

REFERENCES

[1] Mankiw, N. G., "*Principles of macroeconomics*.",(7th ed.), Cengage Learning, United States of America, 2014.

[2] Szabo, N., "*Trusted Third Parties Are Security Holes* [online]*"*, White Paper, 2005, http://szabo.best.vwh.net/ttps.html [11 February 2016].

[3] Nakamoto, S., "*Bitcoin: A peer-to-peer electronic cash system* [online]", 2008, https://bitcoin.org/bitcoin.pdf [11 February 2016].

[4] The World Bank, "*Market capitalization of listed domestic companies (current US$)*[online]", http://data.worldbank.org/indicator/CM.MKT.LCAP.CD [9 April 2016].

[5] Ben-Sasson, E., et al., "*Zerocash: Decentralized anonymous payments from bitcoin"*, In: Security and Privacy (SP), 2014 IEEE Symposium on. IEEE, 459-474, 2014.

[6] Miers, I. et al., "*Zerocoin: Anonymous distributed e-cash from bitcoin"*, In:Security and Privacy (SP), 2013 IEEE Symposium on. IEEE, 397-411, 2013.

[7] King, S., & Nadal, S., "*Ppcoin: Peer-to-peer crypto-currency with proof-of-stake"*, self-published paper, 19 August 2012.

[8] Schwartz, D., Noah Y. & Britto A., "*The Ripple protocol consensus algorithm* [online]", Ripple Labs Inc White Paper, v5, 2014, https://ripple.com/files/ripple_consensus_whitepaper.pdf [12 February 2016].

[9] Wood, G., "*Ethereum: A secure decentralised generalised transaction ledger"*, Ethereum Project Yellow Paper, 2014.

[10] *MinKiz* [online], https://minkiz.co/coin [12 February 2016].

[11] Rogaway, P., & Shrimpton T., "*Cryptographic hash-function basics: Definitions, implications, and separations for preimage resistance, second-preimage resistance, and collision resistance",* Fast Software Encryption 2004, Springer Berlin Heidelberg, 371-388, 2004.

[12] Bellare, M., & Kohno, T., "*Hash function balance and its impact on birthday attacks*", In: Advances in Cryptology-Eurocrypt 2004, Springer Berlin Heidelberg, 401-418, 2004.

[13] Merkle, R. C., "*A digital signature based on a conventional encryption function"*, In: Advances in Cryptology—CRYPTO'87. Springer Berlin Heidelberg, 369-378. 1987.

[14] Merkle, R. C., "*Method of providing digital signatures*" U.S. Patent No. 4,309,569, 5 Jan. 1982.

[15] Okupski, K., "*Bitcoin Developer Reference* [online]", http://enetium. com/resources/Bitcoin. pdf, 2014, [14 February 2016].

[16] Goldwasser, S., Micali S. & Rackoff C., "*The knowledge complexity of interactive proof systems"*, SIAM Journal on computing 18.1, 186-208, 1989.

[17] Quisquater, J. et al., "*How to explain zero-knowledge protocols to your children",* Advances in Cryptology—CRYPTO'89 Proceedings, Springer Berlin/Heidelberg, 628-631, 1990.

[18] Chaum, D., "*Blind signatures for untraceable payments". In: Advances in cryptology,* Springer US, 199-203,1983.

[19] Chaum, D., Amos Fiat, and Moni Naor., "*Untraceable electronic cash"*, Proceedings on Advances in cryptology, Springer-Verlag New York, Inc., 319-327, 1990.

[20] Camenisch, J., Hohenberger, S., & Lysyanskaya, A., "*Compact e-cash"*, In Advances in Cryptology–EUROCRYPT 2005, Springer Berlin Heidelberg, 302-321, 2015.

[21] Rivest, R. L.& Shamir A., "*PayWord and MicroMint: Two simple micropayment schemes"*, Security Protocols. Springer Berlin Heidelberg, 69-87, 1996.

[22] Manasse, M. S., "*Millicent electronic microcommerce"*, Digital Equipment Corp, 1995.

[23] Schoenmakers, B., "*Security aspects of the EcashTM payment system"*, Lecture notes in computer science, 338-352, 1998.

[24] Goldschlag, D. M. & Stuart G. S., "*Publicly verifiable lotteries: Applications of delaying functions"*, Financial Cryptography, Springer Berlin Heidelberg, 214-226, 1998.

[25] Rivest, R. L., "*Peppercoin micropayments"*, Financial Cryptography, Springer Berlin Heidelberg, 2(8), 2004.

[26] Jakobsson, M., and Ari J., "*Proofs of work and bread pudding protocols"*, Secure Information Networks, Springer US, 258-272, 1999.

[27] Dwork, C., & Moni N., "*Pricing via processing or combatting junk mail"*, Advances in Cryptology—*CRYPTO'92*, Springer Berlin Heidelberg, 139-147, 1992.

[28] Back, A., "*Hashcash-a denial of service counter-measure* [online]", http://www.hashcash.org/papers/hashcash.pdf [10 February 2016].

[29] Back, A., "*A partial hash collision based postage scheme* [online]", http://www. hashcash. org/papers/announce.txt [10 February 2016].

[30] Sander, T. & Ta-shma, A., "*Auditable, anonymous electronic cash"*, In: Advances in Cryptology—CRYPTO'99, Springer Berlin Heidelberg, 555-572, 1999.

[31] Dai, W., "*b-money* [online]", 1998, http://www.weidai.com/bmoney.txt [11 February 2016].

[32] Szabo, N., "*Formalizing and securing relationships on public networks"*, First Monday, 2(9), 1997.

[33] Szabo, N., *Bit Gold* [online], http://unenumerated.blogspot.com.tr/2005/12/bit-gold.html [13 February 2016].

[34] *Blockchain.info Genesis Block* [online], https://blockchain.info/block/000000000019d6689c085ae165831e934ff763ae46a2a6c172b3f1b60a8ce26f [12 February 2016].

[35] Bitcointalk, "*List of Major Bitcoin Heists, Thefts, Hacks, Scams, and Losses* [online]", https://bitcointalk.org/index.php?topic=576337 [13 February 2016]

[36] Decker, C., & Wattenhofer, R., "*Bitcoin transaction malleability and MtGox"*, Computer Security-ESORICS 2014, Springer International Publishing, 313-326, 2014.

[37] Garber, L., "*Government Officials Disrupt Two Major Cyberattack Systems"*, 16-16, 2014.

[38] Christin, N., "*Traveling the Silk Road: A measurement analysis of a large anonymous online marketplace"*, Proceedings of the 22nd international conference on World Wide Web, International World Wide Web Conferences Steering Committee, 213-224, 2013.

[39] *Litecoin.org* [online], https://litecoin.org/ [12 February 2016].

[40] King, S., "*Primecoin Whitepaper* [online]", http://primecoin.io/bin/primecoin-paper.pdf [30 April 2016].

[41] Namecoin Team, NameCoin.info [online], https://namecoin.info/ [12 February 2016].

[42] Duffield, E. & Hagan K., "*Darkcoin: PeertoPeer CryptoCurrency with Anonymous Blockchain Transactions and an Improved ProofofWork System"*, 2014.

[43] Britto, A., Schwartz, D., & Fugger, R., Ripple [online], https://ripple.com/, 2012. [9 April 2016].

[44] Gray, M., "*Introducing: Ethereum Blockchain as a Service (EthBaaS)* [online]", https://devcon.ethereum.org/slides/ethBaas_gray.pdf [9 April 2016].

[45] Rosenfeld, M., "*Overview of colored coins",* 2012.

[46] Bentov, I. et al., "*Proof of Activity: Extending Bitcoin's Proof of Work via Proof of Stake",* [Extended Abstract], ACM SIGMETRICS Performance Evaluation Review, 42(3), 34-37, 2014.

[47] CryptoNote Website, https://cryptonote.org/ [9 April 2016].

[48] Van Saberhagen, N., "*Cryptonote v 2."*, 2013, https://cryptonote. org/whitepaper. pdf [10 April 2016].

[49] Bonneau, J., et al. "*SoK: Research perspectives and challenges for Bitcoin and cryptocurrencies"*, Security and Privacy (SP), 2015 IEEE Symposium on. IEEE, 104-121, 2015.

[50] Eastlake, D., & Hansen T., "*US secure hash algorithms (SHA and SHA-based HMAC and HKDF)"*, RFC 6234, 2011.

[51] *Tarsnap Scrypt* [online], http://www.tarsnap.com/scrypt.html [13 February 2016].

[52] Dobbertin, H., Bosselaers, A., & Preneel, B., "*RIPEMD-160: A strengthened version of RIPEMD*", In: *Fast Software Encryption*, Springer Berlin Heidelberg, 71-82,1996.

[53] Eyal, I., & Sirer, E. G., "*Majority is not enough: Bitcoin mining is vulnerable"*, Financial Cryptography and Data Security, Springer Berlin Heidelberg, 436-454, 2014.

[54] Kroll, J. A., Davey I. C., & Felten E.W., "*The economics of Bitcoin mining, or Bitcoin in the presence of adversaries",* Proceedings of WEIS, Vol. 2013, 2013.
[55] "*Blockchains are war* [online]", http://junseth.com/post/119882298052/blockchains-are-war [16 February 2016].
[56] Kaskaloglu, K, "*Near zero Bitcoin transaction fees Cannot last forever"*, In: The International Conference on Digital Security and Forensics (DigitalSec2014), 91-99, 2014.
[57] BitcoinWiki, "*Elliptic Curve Digital Signature Algorithm*[online]", https://en.bitcoin.it/wiki/Elliptic_Curve_Digital_Signature_Algorithm [14 February 2016].
[58] Barker, E., et al., "*Nist special publication 800-57"*, NIST Special Publication 800.57, 1-142, 2007.
[59] Sullivan, N., "*ECDSA: The digital signature algorithm of a better internet* [online]", https://blog.cloudflare.com/ecdsa-the-digital-signature-algorithm-of-a-better-internet/ [15 February 2016].
[60] Vaudenay, S., *"The Security of DSA and ECDSA"*, Public Key Cryptography—PKC 2003, Springer Berlin Heidelberg, 309-323, 2003.
[61] Gennaro, R., Goldfeder S., & Narayanan, A., "*Threshold-optimal DSA/ECDSA signatures and an application to Bitcoin wallet security"*, IACR Cryptology ePrint Archive 2016, 2016.
[62] BitcoinWiki, "*Address* [online]", https://en.bitcoin.it/wiki/Address [16 February 2016].
[63] Penard, W. ,& Werkhoven, T.V., "*On the secure hash algorithm family* [online]", Utrecht University, http://www. staff. science. uu. nl/~ werkh108/docs/study/Y5_07_08/infocry/project/Cryp08. pdf, 2008, [13 February 2016].
[64] NIST Special Publication (SP) 800-107, "*Recommendation for Applications Using Approved Hash Algorithms"*, (Revised), (Draft) September 2011.
[65] Blockchain.info, "*Bitcoin Blockchain Height 396826* [online]", https://blockchain.info/en/block/000000000000000085b7115102dd61a526a1884e04100767382463398a8698e, [13 February 2016].

[66] BitcoinWiki [online], https://en.bitcoin.it/wiki/Block_hashing_algorithm [14 February 2016].
[67] Rosenfeld, M., "*Analysis of Bitcoin pooled mining reward systems"*, arXiv preprint arXiv:1112.4980 , 2011.
[68] Eyal, I., "*The miner's dilemma"*, In: Security and Privacy (SP), 2015 IEEE Symposium on IEEE, 89-103, 2015.
[69] BitcoinWiki, "*Pooled Mining* [online]", https://en.bitcoin.it/wiki/Pooled_mining [16 February 2016].
[70] BitcoinWiki, "*Eligius* [online]", https://en.bitcoin.it/wiki/Eligius [16 February 2016].
[71] BitcoinWiki, "*Difficulty* [online]", https://en.bitcoin.it/wiki/Difficulty [16 February 2016].
[72] Blockexplorer Bitcoin Block Difficulty, https://blockexplorer.com/api/status?q=getDifficulty[16 February 2016].
[73] Cohen, B., "*Incentives build robustness in BitTorrent"*, Workshop on Economics of Peer-to-Peer systems, Vol. 6., 68-72, 2003.
[74] Narayanan, A., Bonneau, J., Felten, E., Miller, A., Goldfeder, S., "*Bitcoin and Cryptocurrency Technologies Draft* ", 2016.
[75] BitcoinWiki, "*Scalability* [online]", https://en.bitcoin.it/wiki/Scalability [16 February 2016].
[76] Tschorsch, F., & Scheuermann, B., "*Bitcoin and Beyond: A Technical Survey on Decentralized Digital Currencies*," IACR Cryptology ePrint Archive2015, 2015.
[77] Donet, J.A. D., Perez-sola, C., & Herrera-Joancomartí, J., "*The bitcoin p2p network"*, In: Financial Cryptography and Data Security, Springer Berlin Heidelberg, 87-102, 2014.
[78] Bitcoin Website, "*Developer Examples* [online]", https://bitcoin.org/en/developer-examples [16 February 2016].
[79] Todd, P., "*Soft forks are safer than hard forks* [online]", https://petertodd.org/2016/soft-forks-are-safer-than-hard-forks [16 February 2016].
[80] Hearn, M., "*On consensus and forks What is the difference between a hard and soft fork?* [online]", https://medium.com/@octskyward/on-consensus-and-forks-c6a050c792e7#.47yryp7l2 [16 February 2016].

[81] Sander, T. & Ta-shma, A., "*Auditable, anonymous electronic cash"*, In: Advances in Cryptology—CRYPTO'99, Springer Berlin Heidelberg, 555-572, 1999.

[82] Schnorr, C. P., "*Efficient signature generation by smart cards"*, Journal of cryptology 4(3), 161-174, 1991.

[83] Blum, M. et.al., "*Non-interactive zero-knowledge and its applications"*, In: Proceedings of the twentieth annual ACM symposium on Theory of computing, ACM, 103-112, 1988.

[84] Johnson, A., "*First Implementation of Zerocoin Released: Introducing Moneta* [online]", Bitcoin News (Bitcoin.com) (18 December 2015), https://news.bitcoin.com/first-implementation-zerocoin-released-introducing-moneta/ [26 February 2016].

[85] Zerocash Team, "*How Zerocash works* [online]", http://zerocash-project.org/how_zerocash_work [28 February 2016].

[86] Zagone R., & Aranda D., "*The 'Ripple' Effect: Why an Open Payments Infrastructure Matters"*, http://www.cgap.org/blog/%E2%80%98ripple%E2%80%99-effect-why-open-payments-infrastructure-matters [13 March 2016].

[87] Cohen, D., Schwartz, D., & Britto, A., "*The Ripple Ledger Consensus Process* [online]", https://ripple.com/knowledge_center/the-ripple-ledger-consensus-process/ Retrieved [13 April 2016].

[88] RippleWiki, "*Federation protocol* [online]", https://wiki.ripple.com/Federation_protocol [30 April 2016].

[89] Armknecht, F., et al., "*Ripple: Overview and Outlook"*, In: Trust and Trustworthy Computing, Springer International Publishing, 163-180, 2015.

[90] Bitcointalk, "*Proof of stake instead of proof of work* [online]", https://bitcointalk.org/index.php?topic=27787.0 [17 March 2016].

[91] Dash source code, "*dashpay/dash* [online]", https://github.com/dashpay/dash Retrieved [16 April 2016].

[92] DASH, *"Ninja -Masternodes Monitoring* [online]", https://dashninja.pl/ [16 April 2016].

[93] Reid, F., & Harrigan M., "*An analysis of anonymity in the bitcoin system"*, Springer New York, 2013.

[94] Narayanan, A., & Shmatikov V., "*De-anonymizing social networks"*, Security and Privacy, 2009 IEEE Symposium on IEEE, 173-187, 2009.

[95] Ron, D., & Shamir A., "*Quantitative analysis of the full bitcoin transaction graph"*, Financial Cryptography and Data Security, Springer Berlin Heidelberg, 6-24, 2013.

[96] Meiklejohn, S. et al., "*A fistful of bitcoins: characterizing payments among men with no name"s*, In: Proceedings of the 2013 conference on Internet measurement conference, ACM, 127-140, 2013.

[97] Biryukov, A. et. al., "*Deanonymisation of clients in Bitcoin P2P network"*, In: Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security, ACM, 15-29, 2014.

[98] Androulaki, E., et al., "*Evaluating User Privacy in bitcoin"*, Financial Cryptography and Data Security, Springer Berlin Heidelberg, 34-51, 2013.

[99] "*Protect your privacy by Bitcoin* [online]", https://bitcoin.org/en/protect-your-privacy [20 February 2016].

[100] Bonneau, J., et al., "*Mixcoin: Anonymity for Bitcoin with accountable mixes"*, Financial Cryptography and Data Security,Springer Berlin Heidelberg, 486-504, 2014.

[101] Maxwell, G., "*CoinJoin: Bitcoin privacy for the real world* [online]",2013, Bitcoin Forum, https://bitcointalk.org/index.php?topic=279249.0 [20 February 2016].

[102] "*What is traffic fingerprinting and how it is used to attack tor network* [online]", http://forcebytes.com/technology-tech/web-social/what-is-traffic-fingerprinting-and-how-it-is-used-to-attack-tor-network/ [21 February 2016].

[103] Biryukov, A., & Pustogarov, I., "*Bitcoin over Tor isn't a good idea"*, Security and Privacy (SP), 2015 IEEE Symposium on IEEE, 122-134, 2015.

[104] Franco, P., "*Understanding Bitcoin: Cryptography, engineering and economics"*, John Wiley & Sons, 141-141, 2014.

[105] BitcoinWiki, "*Thin client security* [online]", https://en.bitcoin.it/wiki/Thin_Client_Security [21 February 2016].

[106] Andresen, G., & Hearn, M., "*BIP 070* [online]", https://github.com/bitcoin/bips/blob/master/bip-0070.mediawiki [21 February 2016].
[107] Diaz , D., & Duffield E., "*Dash: A PrivacyCentric CryptoCurrency* [online]", Dash Whitepaper, https://www.dash.org/wp-content/uploads/2015/04/Dash-WhitepaperV1.pdf [28 February 2016].
[108] Greenberg, A., "*Bitcoin's nefarious cousin Darkcoin is booming"*, http://www.wired.co.uk/news/archive/2014-05/22/darkcoin-is-booming [28 February 2016].
[109] Ross, V., "*Zcash Launches in Alpha* [online]", https://www.cryptocoinsnews.com/zcash-rivals-bitcoin/ [27 February 2016].
[110] Thomas P., "FBI seizes ‘Silk Road' black market domain, arrests owner [online]", https://www.rt.com/usa/silk-road-bitcoin-shut-650/ [20 February 2016].
[111] Christin, N., "*Traveling the Silk Road: A measurement analysis of a large anonymous online marketplace"*, In: Proceedings of the 22nd international conference on World Wide Web, International World Wide Web Conferences Steering Committee, 213-224, 2013.
[112] Greenberg, A., "*An Interview With A Digital Drug Lord: The Silk Road’s Dread Pirate Roberts (Q&A)*[online]", 2013. http://www.forbes.com/sites/andygreenberg/2013/08/14/an-interview-with-a-digital-drug-lord-the-silk-roads-dread-pirate-roberts-qa/#59e30b4c65e7 [21 February 2016].
[113] Christin, N., "*Traveling the Silk Road: A measurement analysis of a large anonymous online marketplace"*, Proceedings of the 22nd international conference on World Wide Web, International World Wide Web Conferences Steering Committee, 2013.
[114] Greenberg, A.,"*Dark Wallet’ Is About to Make Bitcoin Money Laundering Easier Than Ever* [online]", http://www.wired.com/2014/04/dark-wallet/ [21 February 2016].
[115] Moser, M., Bohme, R., & Breuker, D., "*An inquiry into money laundering tools in the Bitcoin ecosystem"*, In: eCrime Researchers Summit (eCRS), IEEE, 1-14, 2013.

[116] "*Ten arrested in Netherlands over bitcoin money-laundering allegations* [online]",http://www.theguardian.com/technology/2016/jan/20/bitcoin-netherlands-arrests-cars-cash-ecstasy [22 February 2016].
[117] The UK Government HM Treasury, "*UK national risk assessment of money laundering and terrorist financing* [online]", https://www.gov.uk/government/uploads/system/uploads/attachment_data/file/468210/UK_NRA_October_2015_final_web.pdf [22 February 2016].
[118] Bitcoin Forum, "*New Attack Vector* [online]", https://bitcointalk.org/index.php?topic=8392.msg122410 [1 March 2016].
[119] Wuille, P., "*BIP62* [online]", 2014, https://github.com/bitcoin/bips/blob/master/bip-0062.mediawiki, [2 March 2016].
[120] Shirriff, K., "*Bitcoin transaction malleability: looking at the bytes* [online]", http://www.righto.com/2014/02/bitcoin-transaction-malleability.html [1 March 2016].
[121] Blockchain [online], https://blockchain.info/rawtx/ed5d19731b6cbfd43494715a2f21538dc191671e076eda750d08ab1e2bcf7125?scripts=true [1 March 2016].
[122] Decker, C.& Wattenhofer R., "*Bitcoin transaction malleability and MtGox*", Computer Security-ESORICS 2014, Springer International Publishing, 313-326, 2014.
[123] Vigna P., "*5 Things About Mt. Gox's Crisis* [online]", http://blogs.wsj.com/briefly/2014/02/25/5-things-about-mt-goxs-crisis/ [1 March 2016].
[124] Buterin, V., "*Transaction Malleability: MtGox's Latest Woes* [online]", https://bitcoinmagazine.com/articles/transaction-malleability-mtgoxs-latest-woes-1392068966 [1 March 2016].
[125] McMillian, R., "*The Inside Story of Mt. Gox, Bitcoin's $460 Million Disaster* [online]", http://www.wired.com/2014/03/bitcoin-exchange/ [1 March 2016]
[126] Wuille, P., "*BIP 66 Strict DER signatures* [online]", https://github.com/bitcoin/bips/blob/master/bip-0066.mediawiki [2 March 2016]
[127] "*Re: New research proves: MtGox bitcoins NOT stolen using transaction malleability*[online]", BitcoinTalk,

https://bitcointalk.org/index.php?topic=537772.msg5957448#msg5957448 [2 March 2016].

[128] Andrychowicz, M. et al., "*How to deal with malleability of bitcoin transactions*", arXiv preprint arXiv:1312.3230, 2013.

[129] 8btc_news, "*Why upgrade to 8MB but not 20MB*? [online]", https://www.reddit.com/r/Bitcoin/comments/3a0n4m/why_upgrade_to_8mb_but_not_20mb/ [9 March 2016].

[130] Bitcointrail [online], https://www.blocktrail.com/BTC/blocks/1 [6 March 2016]

[131] Bitcointrail [online], https://blog.blocktrail.com/2015/08/miners-block-size-vote-explained/ [6 March 2016].

[132] Bitcointrail [online], https://www.blocktrail.com/BTC/pools?resolution=1y [6 March 2016].

[133] Torpey, K., "*6 Proposals for Increasing the Bitcoin Block Size Limit* [online]", https://www.coingecko.com/buzz/6-proposals-increasing-bitcoin-block-size-limit [6 March 2016].

[134] Bitcoin.org, "*Hard Fork Policy* [online]", https://bitcoin.org/en/posts/hard-fork-policy [6 March 2016].

[135] Hearn, M., "*On consensus and forks What is the difference between a hard and soft fork?* [online]", https://medium.com/@octskyward/on-consensus-and-forks-c6a050c792e7#.4wsxz835w [6 March 2016].

[136] Bitcoin Roundtable, "*Bitcoin Roundtable Consensus* [online]", https://medium.com/@bitcoinroundtable/bitcoin-roundtable-consensus-266d475a61ff#.8tobb7qew, [6 March 2016].

[137] Poon, J. & Thaddeus D., "*The bitcoin lightning network: Scalable off-chain instant payments* [online]", Technical Report (draft), 2015, https://lightning.network, [6 March 2016].

[138] Perez,Y. B., "*Could the Bitcoin Lightning Network Solve Blockchain Scalability?* [online]", http://www.coindesk.com/could-the-bitcoin-lightning-network-solve-blockchain-scalability/ [7 March 2016].

[139] Sompolinsky, Y. & Zohar A., "*Accelerating Bitcoin's Transaction Processing. Fast Money Grows on Trees, Not Chains*", IACR Cryptology ePrint Archive, 2013.

[140] Eyal, I. et al., "*Bitcoin-ng: A scalable blockchain protocol*", arXiv preprint arXiv:1510.02037, 2015.

[141] Ripple Forum, "*Maximum txn per second rate on current hardware for network* [online]", https://forum.ripple.com/viewtopic.php?f=2&t=4780 [12 March 2016]

[142] Schwartz, D., "*Is Ripple's distributed exchange scalable?* [online]", http://bitcoin.stackexchange.com/questions/9701/is-ripples-distributed-exchange-scalable [13 March 2016].

[143] Schwartz, D., "*Is Ripple.com scalable?* [online]", http://bitcoin.stackexchange.com/questions/8976/is-ripple-com-scalable [13 March 2016].

[144] Duffield, E., "*Dash v13 - Evolution Design Overview* [online]", https://www.dash.org/binaries/evo/DashPaper-v13-v1.pdf [17 April 2016]

[145] Gill, R., "*CEX.IO Slow to Respond as Fears of 51% Attack Spread* [online]", http://www.coindesk.com/cex-io-response-fears-of-51-attack-spread/ [30 March 2016].

[146] Sirer, E. G., "*Bitcoin Runs On Altruism* [online]", http://hackingdistributed.com/2015/12/22/bitcoin-runs-on-altruism/ [30 March 2016].

[147] Eyal, I., & Sirer, E. G., "*How to Disincentivize Large Bitcoin Mining Pools* [online]", http://hackingdistributed.com/2014/06/18/how-to-disincentivize-large-bitcoin-mining-pools/ [31 March 2016].

[148] Cawrey, D., "*Are 51% Attacks a Real Threat to Bitcoin?* [online]", http://www.coindesk.com/51-attacks-real-threat-bitcoin/ [30 March 2016].

[149] Bastiaan, M., "*Preventing the 51%-Attack: a Stochastic Analysis of Two Phase Proof of Work in Bitcoin* [online]", 2015, *http://referaat.cs.utwente.nl/conference/22/paper/7473/preventingthe-51-attack-a-stochastic-analysis-of-two-phase-proof-of-work-in-bitcoin. pdf*. [30 March 2016].

[150] Rowaihy, H. et al.,"*Limiting sybil attacks in structured p2p networks"*, INFOCOM 2007, 26th IEEE International Conference on Computer Communications, 2596-2600, 2007.

[151] Bissias, G. et al., "*Sybil-resistant mixing for bitcoin"*, Proceedings of the 13th Workshop on Privacy in the Electronic Society, ACM, 149-158, 2014.
[152] Heilman, E. et al., "*Eclipse Attacks on Bitcoin's Peer-to-Peer Network"*, 24th USENIX Security Symposium (USENIX Security 15), 129-144, 2015.
[153] Bitcoincharts, http://bitcoincharts.com/charts/mtgoxUSD#czsg2013-04-03zeg2013-04-05ztgSzm1g10zm2g25zv [14 March 2016].
[154] Buterin, V., "*The Bitcoin Crash: An Examination* [online]", https://bitcoinmagazine.com/articles/the-bitcoin-crash-an-examination-1365911041 [14 March 2016].
[155] Bitcoincharts, http://bitcoincharts.com/charts/mtgoxUSD#rg5zig5-minzczsg2013-04-10zeg2013-04-13ztgSzm1g10zm2g25zv [14 March 2016].
[156] Eyal, I., & Sirer, E.G., "*Bitcoin-NG: A Secure, Faster, Better Blockchain* [online]", http://hackingdistributed.com/2015/10/14/bitcoin-ng/ [29 March 2016].
[157] BitcoinWiki, "*Denial of service* [online]", https://en.bitcoin.it/wiki/Weaknesses#Denial_of_Service_.28DoS.29_attacks [13 March 2016].
[158] Vasek, M., et al., "*The Bitcoin Brain Drain: A Short Paper on the Use and Abuse of Bitcoin Brain Wallets*", Financial Cryptography and Data Security, Lecture Notes in Computer Science, Springer, 2016.
[159] Bitcoin website, "*Securing your wallet* [online]", https://bitcoin.org/en/secure-your-wallet [14 March 2016].
[160] Pham, P.T., & Lee, S., "*Anomaly Detection in Bitcoin Network Using Unsupervised Learning Methods* [online]", http://cs229.stanford.edu/proj2014/Phillip%20Pham,Steven%20Li,%20Anomaly%20Detection%20in%20Bitcoin%20Network%20Using%20Unsupervised%20Learning%20Methods.pdf [30 April 2016].
[161] Schwartz, D., "*What is the Ripple equivalent of the "51% attack?*[online]", http://bitcoin.stackexchange.com/questions/10181/what-is-the-ripple-equivalent-of-the-51-attack [17 April 2016].
[162] Rothstein, A., "*How Much Electricity Does Bitcoin Use?* [online]", https://medium.com/@interdome/how-much-electricity-does-bitcoin-use-

c350bd84c64e#.6arxydv50 https://medium.com/@interdome/how-much-electricity-does-bitcoin-use-c350bd84c64e#.yofn1mves [29 March 2016].
[163] Global Energy Statistical Yearbook 2015, https://yearbook.enerdata.net/#electricity-domestic-consumption-data-by-region.html [29 March 2016].
[164] Gimein, M., "*Virtual Bitcoin Mining Is a Real-World Environmental Disaster*", *Bloomberg Business*, 2013.
[165] Swartz, A., "*Squaring the Triangle: Secure, Decentralized, Human-Readable Names* [online]", http://www.aaronsw.com/weblog/squarezooko [30 April 2016].
[166] Loibl, A., "*Namecoin, Seminars FI/IITM SS*", Network Architectures and Services, August 2014.
[167] Ethereum Team, "*Ethereum Whitepaper* [online]", https://github.com/ethereum/wiki/wiki/White-Paper [30 April 2016].
[168] Buterin, V., "*Vitalik's Research and Ecosystem Update* [online]", 2015, https://blog.ethereum.org/2015/10/18/vitaliks-research-and-ecosystem-update/, [3 April 2016].
[169] Buterin, V., "*Long-Range Attacks: The Serious Problem With Adaptive Proof of Work*[online]", https://blog.ethereum.org/2014/05/15/long-range-attacks-the-serious-problem-with-adaptive-proof-of-work/ [3 April 2016]
[170] Github, "*Ethereum/Wiki* [online]", https://github.com/ethereum/wiki/wiki/Problems, [3 April 2016].
[171]] Buterin, V., "*The P + epsilon Attack* [online]", 2015, https://blog.ethereum.org/2015/01/28/p-epsilon-attack/ [3 April 2016].
[172] Colored Coin Github, https://github.com/Colored-Coins/Colored-Coins-Protocol-Specification/wiki/Coloring%20Scheme [25 April 2016].
[173] OpenAssets Github, https://github.com/OpenAssets/open-assets protocol/blob/master/specification.mediawiki, [24 April 2016].
[174] Litecoin, "*Mining hardware comparison* [online]", https://litecoin.info/Mining_hardware_comparison, [19 March 2016].

[175] Schwen, D., "*Zcash, an Untraceable Bitcoin Alternative, Launches in Alpha* [online]", http://www.wired.com/2016/01/zcash-an-untraceable-bitcoin-alternative-launches-in-alpha/ [28 February 2016].

[176] King, S., "*Primecoin: Cryptocurrency with prime number proof-of-work"*, 2013.

[177] Gennaro, R. et al., "*Quadratic Span Programs and Succinct NIZKs without PCPs*", In: EUROCRYPT, 626-645, 2013

[178] PayPal webpage https://web.archive.org/web/20141226073503/https://www.paypal-media.com/about [19 March 2016]

[179] Buterin, V., "*Selfish Mining: A 25% Attack Against the Bitcoin Network* [online]", https://bitcoinmagazine.com/articles/selfish-mining-a-25-attack-against-the-bitcoin-network-1383578440 [16 March 2016].

[180] Andresen, G., "*BIP16* [online]", https://github.com/bitcoin/bips/blob/master/bip-0016.mediawiki [16 February 2016].

[181] Groth, J., "*Short pairing-based non-interactive zero-knowledge arguments"*, Advances in Cryptology-ASIACRYPT 2010, Springer Berlin Heidelberg, 321-340. 2010.

[182] [BCN] Bytecoin., "*Secure, private, untraceable since 2012* [online]", https://bitcointalk.org/index.php?topic=512747.0 [10 April 2016].

[183] Bitansky, N. et al., "*Succinct non-interactive arguments via linear interactive proofs*", Theory of Cryptography, Springer Berlin Heidelberg, 315-333, 2013.

[184] Buterin, V., "Reddit r/Ethereum [online]", https://www.reddit.com/r/ethereum/comments/3o9ru0/transaction_malleability_does_ethereum_have_this/cvvazon [3 April 2016].

[185] Gehring, B., "*What is the Ripple Protocol?* [online]", https://ripple.com/knowledge_center/what-is-a-protocol-how-does-ripple-fit-in-2/ [12 April 2016].

[186] Douceur, J. R., "*The sybil attack." Peer-to-peer Systems*", Springer Berlin Heidelberg, 251-260, 2002.

[187] Vega, D., "*Peercoin: 5 Fast Facts You Need to Know* [online]", http://heavy.com/tech/2013/12/what-is-peercoin-cryptocurrency/ [9 April 2016].

[188] Bertoni, G. et al., "*Keccak sponge function family main document*", Submission to NIST (Round 2)(3), 2009

[189] User with nick TaoOfSatoshi, "*Tao's Masternode Setup Guide For Dummies*[online]", https://dashtalk.org/threads/taos-masternode-setup-guide-for-dummies.2680/ [16 April 2016].

[190] Goldreich, O., "*Foundations of cryptography: volume 1, basic applications*", Cambridge University Press, Israel, 2001.

[191] Benaloh, J. & De Mare, M., "*One-way accumulators: A decentralized alternative to digital signatures*", In: Advances in Cryptology—EUROCRYPT'93, Springer Berlin Heidelberg, 274-285, 1993.

[192] "*Transaction Malleability* [online]", https://wiki.ripple.com/Transaction_Malleability [17 April 2016].

[193] Wilcox-O'Hearn, Z., "*Names: Distributed." Secure, Human-Readable: Choose Two*", 2001.

[194] Light, J., "*Scaling Cryptocurrency With Ripple* [online]", https://letstalkbitcoin.com/scaling-cryptocurrency-with-ripple [12 March 2016].

[204] Clark, C., "*Bitcoin Internals - A technical guide to Bitcoin"*, (KindleEdition), 2013.

[195] Ben-Sasson, E. et al., "*Secure sampling of public parameters for succinct zero knowledge proofs*", In: Security and Privacy (SP), 2015 IEEE Symposium on IEEE, 287-304, 2015.

[196] Garzik, J., "*Making Decentralized Economic Policy BIP 100 - Theory and Discussion* [online]", v0.8.1 - draft, http://gtf.org/garzik/bitcoin/BIP100-blocksizechangeproposal.pdf [6 March 2016].

[197] Lipmaa, H., "*Progression-free sets and sublinear pairing-based non-interactive zero-knowledge arguments*", Theory of Cryptography. Springer Berlin Heidelberg, 169-189, 2012.

[198] Buterin, V., "*SchellingCoin: A Minimal-Trust Universal Data Feed* [online]", https://blog.ethereum.org/2014/03/28/schellingcoin-a-minimal-trust-universal-data-feed/ [3 April 2016].

[199] Okamoto, T., & Kazuo O., "*Universal electronic cash*", Advances in Cryptology—CRYPTO'91, Springer Berlin Heidelberg, 324-337, 1991.

[200] Lee, T., "*Bitcoin's Volatility Is A Disadvantage, But Not A Fatal One* [online]", http://www.forbes.com/sites/timothylee/2013/04/12/bitcoins-volatility-is-a-disadvantage-but-not-a-fatal-one/#2ee6c048635e [30 March 2016].

RESUME

CAVİDAN YAKUPOĞLU

PERSONAL INFORMATION

E-mail: c.yakupoglu28@gmail.com, cyakupoglu@ybu.edu.tr

Birthday: 01.10.1989

Birth Place: Kastamonu, TURKEY

Gender: Female

EDUCATION

Graduate: Yıldırım Beyazıt University, ANKARA**,** Computer Engineering, Master Student (2014 January- Present)

Interest: CryptoCurrency, Cryptographic Protocols, Information Security, Android Security

Master Thesis: Comparison of Bitcoin and Altcoins

Graduate: İstanbul Şehir University, iSTANBUL, Electronic and Computer Engineering (English) Master Student, (2013 January – 2014 January)

Interest: Electronic Voting Systems (Scantegrity)

Undergraduate: TOBB University of Economics and Technology, ANKARA Mathematics, 2012

Double Major: TOBB University of Economics and Technology, ANKARA Computer Engineering, 2012 *(Contains all courses of department)*

EMPLOYMENT

- Yıldırım Beyazıt University, ANKARA**,** Computer Engineering Department Research/Teaching Assistant (2014 January- Present)
- İstanbul Şehir University, ISTANBUL, Electronic and Computer Engineering Department Research/Teaching Assistant (2013 January- 2014 January)

LANGUAGES

- **Turkish** – Native language
- **English** – High level reading/writing/listening/speaking
- **French** – Average level reading/writing/listening/speaking