Security Issues in Blockchain-Based Network Applications

Johan Burke
Computer Science Department
Rowan University
Glassboro, NJ
burkew9@students.rowan.edu

***Abstract*—Peer to peer network applications based on distributed ledgers called blockchains have seen continuously increasing use in recent years. In particular, cryptocurrencies are a new method of conducting finance by leveraging the power of peer to peer networks and cryptographic algorithms. Such applications have unique security considerations that must be addressed in order to prevent a multitude of network attacks that have drastic consequences upon being successfully carried out.**

***Index Terms*—blockchain; security; cryptocurrency; peer-to-peer networking**

I. Background

In peer-to-peer networks, users both use and provide a service by sharing part of their own computing power with other members of the network. A key aspect of such networks is that they are decentralized, meaning there is no central control point governing the network; rather, the system's behavior is determined by the actions of the network's peers. Peer to peer networks have been used for several applications, including file sharing, Internet telephony, and video streaming [1].

A. Cryptocurrency

An emerging technology that makes use of peer to peer networking is cryptocurrency. A cryptocurrency provides a system for distribution and generation of digital currency units through the use of cryptographic algorithms and a distributed public ledger called a blockchain. The blockchain is a record of all the transactions of the currency that have ever occurred and contains a chain of blocks that grows as transactions are completed. Each block contains a reference to the previous block as well as the transaction data it describes [2]. One of the first widely circulated cryptocurrencies was Bitcoin, which began circulating in 2009. They are created digitally, and, unlike fiat currency, do not have a central issuing organization such as a government or bank. The coins are minted according to carefully calculated parameters to preserve scarcity value and prevent inflation as well as to allow economical transfers [3].

B. Structure of a Blockchain

A block consists of the block header and the block body. The *block header* includes the following elements:

- Block version: indicates which set of block validation rules to follow
- Merkle Tree Root Hash: the hash value of all transactions in the block
- Time Stamp: current time (seconds) in Universal Time since January 1, 1970
- nBits: the target threshold of a valid block hash
- Nonce: a 4-byte field, which usually starts with 0 and increases for every hash calculation
- Parent Block Hash: a 256-bit hash value that points to the previous block

The *block body* consists of a transaction counter and transactions [4].

C. Types of Blockchains

1) Public blockchains: Many modern public blockchains are open source and based on Proof of Work (PoW) consensus algorithms. Any user can participate in the network, without permission, by downloading the code and running a public node on their public device, validating the network's transactions. Anyone in the world can send transactions

over the network, and, if they are valid, expect to see them included in the blockchain. Examples of public blockchains include Bitcoin, Ethereum, and Litecoin [5].

2) Federated blockchains or consortium blockchains: Federated or consortium blockchains are run under the leadership of a group. Unlike public blockchains, access to network participation and transaction validation is not open to the public. Federated blockchains are faster than public blockchains (thus having higher scaleability), and they provide more transaction privacy. Consortium blockchains are mostly found in the banking sector. A pre-selected set of nodes controls the consensus process: for example, a consortium might consist of 15 financial institutions, where each controls a node. This consortium might require at least 10 of the institutions sign a block for that block to be validated. Read access to the blockchain may be granted to the public, or it might also be limited to the network's participants. An example of a federated blockchain is Corda [5].

3) Private blockchains: In a private blockchain, a single, central organization controls write permissions for the blockchain. Read access may be either public or restricted to arbitrary events. Private blockchains provide the ability to take advantage of blockchain technology by creating groups of participants who can verify transactions internally. Examples of private blockchains include Monax and MultiChain [5].

D. Consensus Methods

Within cryptocurrencies, transactions occur when a payer sends some currency to a payee. Transactions are verified by miners, who validate the transactions and add them to the block chain. Mining prevents users from attempting to double spend. Malicious users might attempt to own multiple nodes to validate illegitimate transactions. Cryptocurrencies address this problem by requiring miners to perform some resource intensive task. The resource intensity makes it expensive for malicious users to create the number of false identities necessary to have more influence than legitimate users. There are several different consensus methods used by cryptocurrencies [2]:

1) *Proof of Work* is a proof scheme that consists of an easily verifiable, resource intensive task whose result indicates that the task has been performed. Bitcoin, the first cryptocurrency, uses a Proof of Work scheme; specifically, it uses a hash algorithm called Hashcash. This algorithm requires miners to find a nonce, which, when hashed in conjunction with the hash of the previous blocks, results in a hash with a specified number of leading zeroes. For Ethereum, the developers developed their own hashing algorithm, EtHash, instead of using a preexisting hash algorithm.
2) *Proof of Stake* is a proof scheme where miners must indicate how much currency, or "stake," they have in the system. For example, BlackCoin uses a process called minting to secure its network. Minting is a Proof of Stake system that validates a transaction in less time than and is independent of Proof of Work.
3) *Proof of Retrievability* is a proof scheme where the miner must show that the data he or she received is intact and can be recovered if necessary. The theoretical Permacoin introduces the new concept of Proof of Retrievability, which involves miners storing some considerably sized useful information and presenting a proof to the verifier that the information exists.

II. Security Issues of Blockchain-Based Applications

It is necessary to address security issues of blockchain based applications, particularly those of cryptocurrencies. Successful attacks against cryptocurrencies can result in millions of dollars of loss as well as compromised devices. For instance, in 2011, Bitcoin exchange Mt. Gox, was hacked, resulting in a price dive in Bitcoin from $17.50 to $0.01 in the span of 30 minutes. In this attack, Mt. Gox users' emails, usernames, and hashed passwords were released on the Internet [6]. This resulted in a loss of the equivalent of about $8.75 million [7]. Several types of attacks exist against cryptocurrencies [2] [8] [9]:

1) *51% attack* An attack on a proof of work based system in which a user controls at least 51% of the network's processing power. This

would allow the user to sustain the longest chain, thereby effectively controlling the system.

2) *Double Spending* An attacker may issue two or more transactions using the same coin, thereby spending more coins than he or she actually has.
3) *Selfish Mining* An attacker increases his or her relative mining share in the blockchain by withholding certain mined blocks and only gradually publishing them. Using such an approach, a miner who originally has 33% of the mining power can have effectively 50% of the mining power.
4) *Sybil Attack* An attacker takes on multiple identities and uses them to gain an influence that is large enough to outnumber legitimate nodes. Such an attack can manifest as a Denial of Service attack by not relaying transaction information, effectively disconnecting legitimate nodes from the network, or as double spending by selectively relaying transaction information
5) *Eclipse Attack* An attacker with a sufficient number of IP addresses monopolizes all connections to and from a victim node, resulting in double spending, selfish mining, and adversarial forks

III. Technical Approach

To approach the solution to the problem of security vulnerabilities in blockchain-based applications, a network simulation can be constructed using the OMNeT++ framework for peer-to-peer simulation. Network models will be constructed iteratively, starting from very basic blockchain simulation, to simulating complex blockchain behavior, including each consensus method and attack type [10]. OMNeT++ provides a component-based C++ simulation library and framework and places an emphasis on extensibility and modularity. It also includes an integrated development and a graphical runtime environment to ease simulation development and testing. OMNeT++ has been used in numerous domains, including peer-to-peer network simulations [11]. Blockchains with various consensus methods, attacks, and protections against these attacks are packaged in an OMNeT++ simulation, source code for which can be found at https://github.com/relativeabsolute/CN-Simulation. The simulation is modeled closely off of the source code of Bitcoin Core [12].

A. Components

The OMNeT++ simulation consists of several components, which are interconnected at runtime to simulate the peer to peer network of a blockchain. These components include:

1) POWNode: a subclass of OMNeT++'s simple module class. A POWNode handles the computations that would be done by a single node in a Proof-of-Work based network.

2) POWNetwork: a network that contains a collection of POWNodes. Contrary to traditional OMNeT++ networks, this network does not handle connecting the nodes to each other, since connections are handled at runtime.

3) Network Definition: user specified parameters for the simulation. These include time intervals for scheduled messages, the default set of nodes each node should attempt to connect to at start up, etc.

B. Simplifications

In modeling the behavior of Bitcoin Core, several simplifications can be made. These simplifications ease the development of the simulation and make understanding of the simulation more intuitive. Simplifications employed are as follows:

1) Threading: Bitcoin Core divides tasks into threads, which are subprocesses running within the node's parent process. Implementing threads or using parallel computation would introduce significant overhead to the simulation, and thus threads are modeled using OMNeT++'s concept of self-scheduled messages. Upon initialization, a node schedules several self-messages according to the corresponding parameters specified in the network definition.

2) Routing: OMNeT++ provides the concept of gates, which allow connections between nodes to be constructed easily. Gates provide an abstraction for low level networking procedures, and thus routing, listening to sockets, and other network layer tasks handled by Bitcoin Core do not need to be dealt with by the simulation.

3) Timestamps: Bitcoin Core uses timestamps in several places to check if addresses should be updated, to validate blocks, and more. Timestamps are not used with addresses in the simulation to simplify peer discovery and routing. When time is used, such as in scheduling self-messages, time intervals are scaled down from those used by Bitcoin to ones more appropriate for simulation time, in order to maintain simulation practicality.

C. Steps

Steps run in the simulation can be divided into initialization steps and blockchain steps.

1) Initialization: Initialization of the network involves performing steps necessary to allow nodes to connect to each other and successfully perform transactions. Steps in initialization include:

1) *Reading network parameters* Before attempting to connect to other nodes, each node must read the parameters specified in the network definition. This stage also includes reading any data files if network persistence is desired. Otherwise, each node must use default settings to set up the network for the first time. Handlers for self-messages that are scheduled to be delivered at regular intervals are also connected in this step.
2) *Outbound connections* Non-default nodes initiate outbound connections with their known peers.
3) *Initial messages* Once connections are established, nodes send messages indicating the version of the protocol they are using to their known peers, to let those peers know they are on the network.

2) Blockchain: Once initialization is done, the blockchain is ready to run. The blockchain runs by reading from a user-written schedule file, which contains instructions to initiate transactions, add new nodes to the network, or disconnect existing nodes from the network. This schedule file is also where specific attacks to be run on the network are defined.

IV. Literature Review

A. Solutions to Blockchain Attacks Found in Literature

1) HSMs to safeguard digital keys and protect blockchain ledgers: In his article, O. Boireau writes that several organizations uses hardware security modules (HSMs) to safeguard and manage their digital keys. An HSM is a crypto-processor used to securely generate, protect, and store keys. By using HSMs to protect blockchain ledgers, digital wallets, and applications against hacks, network developers and maintainers can provide a computing environment with the necessary trustworthiness to take full advantage of the blockchain protocol. Carrying out a successful attack against an HSM-enabled network would require administrative privileges, pre-encryption access to data, or physical HSM access. These requirements increase attack difficulty and make the attack unprofitable for a hacker [13].

2) Hash-chain (blockchain) as a distributed timestamp mechanism: Kuo et al. offer a solution to the problem of double spending by requiring that each computational node in the network not only store every transaction for the distribution of transaction verification, but also follow a distributed timestamp mechanism to determine which transactions should be accepted and rejected. Hash-chain is used in Bitcoin: every node maintains a copy of the chain to story every transaction (including timestamps), and each block creator must follow a proof of work mechanism [14].

3) Innovative solution to a 51% attack: O'Leary reports that the team behind the cryptocurrency project Horizen (formerly known as zencash) offer an innovative solution to the problem of 51% attacks. The Horizen team proposes the addition of a delay function to their proof-of-work consensus scheme. The delay function would heavily penalize miners for attempting to conduct a 51% attack. According to Horizen co-founder Rob Viglione, the requirement that a miner produces blocks in secret to conduct a 51% attack means that a delay function would make such an attack prohibitively expensive (up to 10x cost) [15]. The algorithm introduces penalties for delayed blocks (blocks broadcast to the network a long time after the latest block) by modifying the longest-chain rule feature of the

underlying consensus algorithm. As an example, a proposed block that is 5 or more blocks behind the most recent block quadratically increases the amount of blocks a miner needs to produce in order to have transactions accepted in the chain. Thus, the chances of a 51% attack are decreased while only penalizing malicious miners [15].

4) SyMon: Defending large structured P2P systems against a Sybil attack: Jyothi et al. introduce a solution to protect honest peers of large P2P systems against Sybil attacks with a high probability of success. In the proposed system, each peer has an associated non sybil peer known as a SyMon (Sybil Monitor). Each peer is connected with a SyMon dynamically in such a way that the chances of the peer and the SyMon being sybils are very low. The SyMon prevents its peer from targeting honest peers by monitoring the peer's transactions, making it almost impossible for sybils to compromise the system [16].

5) Solutions to double spending: Proposed solutions to double spending include the following:

1) *Countermeasure against double-spending attacks in the fast-pay setting* Podolanko et al. propose a countermeasure against double spending in fast pay settings by introducing enhanced observers (ENHOBS), a system that is a hybrid of observers and the peer alert system. ENHOBS conduct deeper inspections of received transactions, including comparing all outputs and inputs. Upon detection of a double spend, an alert message containing both the incriminating transactions is sent across the P2P network. Upon receipt and verification of the alert, any transactions with the same inputs are dropped immediately from the memory pool.
2) *A solution to the double-spending problem using a peer-to-peer network* Nakamoto, the creator of Bitcoin, proposes the use of a P2P network to solve the double spending problem. The network hashes transactions in an ongoing chain of hash based proof-of-work. The chain thus forms a record that cannot be changed without redoing the proof of work. The longest chain serves a dual purpose of providing proof of the sequence of events witnessed and of providing proof that the events came from the largest pool of CPU power. So long as the majority of computing power is not controlled by nodes cooperating to coordinate an attack, they will generate the longest chain and outpace attackers. Nodes that leave and rejoin the network can simply accept the longest chain as proof of what happened when they were gone [17].
3) *Double-spending prevention for Bitcoin zero-confirmation transactions* Prez-Sol et al. write that zero-confirmation transactions are not protected against double spending attacks, since the mechanism Bitcoin offers for double spend protection relies on the blockchain, in which zero-confirmation transactions are by defintion not yet included. In their paper, the authors propose a double-spending prevention mechanism for Bitcoin zero-confirmation transactions. Their proposed mechanism discourages double spending attempts by forcing private key disclosure in the case of double spending, using special outputs. By detecing double spending attempts, any Bitcoin network user can act as an observer and obtain a reward equal to the price paid by a double spender. The authors' solution benefits scenarios that require fast-payment scenarios such as in-shop purchases or trading platforms, where the transfer of Bitcoin for product or service cannot wait for the transaction to be verified by the blockchain [18].

6) Solutions to selfish mining attacks: Proposed solutions to selfish mining include the following:

1) *ZeroBlock: Preventing selfish mining in Bitcoin* The authors present a new solution called ZeroBlock in their technical report. The scheme prevents block withholding (selfish mining) without using timestamps. In ZeroBlock, a block being held privately by a selfish node longer than the expected time calculated by honest nodes will expire and be rejected by honest nodes. Honest nodes thus never accept chains infested with block withholding when using this scheme. Additionally, an honest node joining a dynamic network can diagnose the correct chain from chains infected with block withholding [19].

2) *A backward-compatible defense against selfish mining in Bitcoin* In their report, the authors note that two policies come into play when determining which blocks receive mining rewards: the fork-resolving policy, which requires that the main chain is the longest chain, and the reward distribution policy, which demands that all blocks in the main chain and no other blocks receive rewards. Changing the latter policy makes the protocol backward-incompatible, and thus the authors attempt to change the fork-resolving policy. The authors replace the original fork-resolving policy (FRP), denoted by length FRP, with a weighted FRP. The weighted FRP asks miners to compare the weight of the chains instead of their length, thereby putting selfish miners into a dilemma: keeping a block secret after a competing block is published means the secret means the secret block does not contribute to the weight of its chain, and publishing the secret block with the competing block means the next honest block gains a higher weight by embedding a proof of seeing the current block. In either case, the selfish miner does not get ahead in the block race with the secret block [20].

7) Eclipse attacks on overlay networks: Threats and defenses: Singh et al. present a new defense against Eclipse attacks that involves nodes anonymously auditing each other's connectivity. Their key observation is that a node that wishes to mount an Eclipse attack must have a higher-than-average node degree. Therefore, an Eclipse attack can be prevented by having nodes choose neighbors whose in-degree is not significantly higher than average. Correct nodes choose neighbors with in- and out-degrees are below a given threshold. Thus, if a node has a significantly higher than average number of links, it is probably mounting an Eclipse attack. Routine auditing of this type leads to the discovery and removal of attackers from neighbor sets of correct nodes [21].

B. Blockchain alternatives

1) Colored Coins: Colored Coins is an advanced application of Bitcoin's blockchain. The author states that the protocol essentially has Bitcoin 2.0 properties through its expansion of the use of Bitcoin's blockchain. The Colored Coins themselves are tokens used to represent real-world assets. For instance, they can prove ownership of cars, real estate, precious metals, etc. and are relatively easy to issue. While not many currently existing cryptocurrencies are Colored Coins, the ones that are provide some interesting features. Colored Coins are most commonly used by companies wishing to host an IPO, because setting up an IPO with Colored Coins can be done in a matter of minutes at a very low cost [22].

2) Tangle: P. Schueffel discusses Tangle, a distributed ledger technology similar to Bitcoin but different in the key sense that it does not require all participants to be in sync regarding all information at all times. Instead, a Tangle is built around the concept of acyclicity; that is, it makes no assumptions about when information reaches which network participant. As such, Tangle does not use conventional blocks but instead requires that nodes approve of two previous transactions when carrying out a new transaction. This means that initiators of new transactions indirectly confirm that a subsection of the Tangle is valid and conforms to the protocol rules [23].

3) Hashgraph: Schueffel also discusses Hashgraph, which uses an entirely different method of sharing information and establishing consensus. One network participant must share all information on transactions with multiple other randomly selected network nodes. The next node then combines the information it receives from the first node with the information received from other participants and add any information about new transactions. The resulting set of aggregated information is then passed on to the next randomly chosen nodes until all participants know the information created at the beginning. Hashgraph is a closed source, patented technology [23].

4) Sidechains: Crosby et al. discuss sidechains, which are alternative blockchains backed by Bitcoins via Bitcoin contract, analogous to how dollars and pounds were once backed by gold. It is theoretically possible to have thousands of sidechains "pegged" to Bitcoin, each with unique characteristics and purposes. All of these sidechains takes advantage of the scarcity and resilience guaranteed

by the Bitcoin blockchain. Conversely, the Bitcoin blockchain can iterate to support additional features for the experimental sidechains once they have been sufficiently tried and tested [24].

C. Blockchain Technology Applications

1) Financial Applications:

1) *Private securities* Crossby et al. write that companies can now directly issue their shares by making use of the blockchain. Shares can be bought and sold over a secondary market that uses a blockchain as its foundation. An example of this usecase is NASDAQ Private Equity: NASDAQ has joined hands with the San Francisco start-up chain.com to implement private equity exchange on top of the blockchain. The product is expected to be fast, traceable, and efficient [24].
2) *Insurance* The blockchain can potentially be used to register any property, allowing the ownership and transaction history to be validated by anyone, especially insurers. An example is Everledger, a company that creates diamond certifications and transaction histories in a permanent ledger using a blockchain [24].

2) Non-Financial Applications:

1) *Smart contracts* Smart contracts are self-executing contracts where the contract's terms being encoded directly into lines of code. The code and the agreements it contains are distributed across a decentralized blockchain network. Smart contracts enable trusted transactions and agreements among anonymous parties without requiring a central authority, legal system, or external enforcement mechanism. The transactions are traceable, transparent, and irreversible [25].
2) *Notary public* The blockchain can be used to verify authenticity of documents without the need for a centralized authority. An example is Crypto Public Notary, where trivial amounts of bitcoins are used to record a file's checksum in the public blockchain [24].
3) *Decentralized proof of existence of documents* Validating signed documents' existence or possession is integral to any legal situation, and blockchain technology can be used to do so in a decentralized fashion. A user simply stores their signature and timestamp associated with a legal document in the blockchain. These can then be validated at anytime using blockchain mechanisms. An example of such a service is Proof of Existence, which simply allows one to anonymously and securely store proof of existence online for any document. The service stores the file's fingerprint and links it to the time of submission [24].
4) *EOS* EOS is a blockchain platform that facilitates decentralized applications' development. EOS provides a ready-made platform for appliations that allow developers a full-featured authentication system. User accounts each have different permission levels and their own locally secured data. A user can also share database access across accounts and locally store user data [26].

3) Biomedical and Health Care Applications: Kuo et al. review blockchain distributed ledger technologies for biomedical and health care applications. The decentralized nature of blockchains make them ideal for applications where independently managed biomedical/health care stakeholders (e.g., hospitals, providers, patients, and payers) wish to collaborate with one another without ceding control to a central management intermediary. Potential applications of blockchain technology in biomedical/health care applications include improving medical record management, enhancing the insurance claim process, accelerating clinical/biomedical research, and advancing biomedical/health care data ledgers [14].

4) Blockchains for Research: J. Van Rossum, discusses potential applications of blockchains for research. Researchers working on a blockchain could create or interact in whatever way and at whatever stage with interactions stored on a single platform. Critical aspects of scholarly communication such as trust, credit, universal access, andwhere requiredanonymity, can be realized and safeguarded in a research focused blockchain [27].

REFERENCES

[1] X. Shen, J. Buford, H. Yu, and M. Akon, Eds., *Handbook of Peer-to-Peer Networking*. New York, NY: Springer, 2010. [Online]. Available: https://link-springer-com.ezproxy.rowan.edu/book/10.1007\%2F978-0-387-09751-0

[2] U. Mukhopadhyay, A. Skjellum, O. Hambolu, J. Oakley, L. Yu, and R. Brooks, "A brief survey of cryptocurrency systems," in *Proc. IEEE Annual Converence on Privacy, Security and Trust (PST'16)*, Auckland, New Zealand, Dec. 2016. [Online]. Available: https://ieeexplore-ieee-org.ezproxy.rowan.edu/stamp/stamp.jsp?tp=\&arnumber=7906988

[3] J. Downes and J. E. Goodman, "Bitcoin," in *Dictionary of Finance and Investment Terms*. Hauppage, New York: Barron's Educational Series, 2014. [Online]. Available: https://search.credoreference.com/content/entry/barronsfin/bitcoin/0

[4] M. Niranjanamurthy, B. N. Nithya, , and S. Jagannatha, "Analysis of blockchain technology: pros, cons and swot," *Cluster Computing*, pp. 1–15, Mar. 2018. [Online]. Available: https://doi.org/10.1007/s10586-018-2387-5

[5] Blockchains & distributed ledger technologies. [Online]. Available: https://blockchainhub.net/blockchains-and-distributed-ledger-technologies-in-general/

[6] M. Karpeles. (2011, Sep.) Clarification of mt. gox compromised accounts and major bitcoin sell-off. Tibanne Co. Ltd.

[7] J. Mick. (2011, Jun.) Inside the mega-hack of bitcoin: the full story. Daily Tech.

[8] A. Gervais, G. Karame, K. Wüst, V. Glykantzis, H. Ritzdorf, and S. Capkun, "On the security and performance of proof of work blockchains," in *Proc. ACM SIGSAC Conference on Computer and Communications Security (CCS'16)*, Vienna, Austria, Oct. 2016.

[9] E. Heilman, A. Kendler, A. Zohar, and S. Goldberg, "Eclipse attacks on bitcoin's peer-to-peer network," in *Proc. USENIX Security Symposium (USENIX Security'15)*, Washington, D.C., Aug. 2015.

[10] (2018) Simulation models. OpenSim Ltd. [Online]. Available: https://omnetpp.org

[11] A. Varga, "Omnet++," in *Modeling and Tools for Network Simulation*. Berlin, Germany: Spring-Verlag, 2010, pp. 35–59.

[12] (2018) Bitcoin. [Online]. Available: https://github.com/bitcoin/bitcoin

[13] O. Boireau, "Securing the blockchain against hackers," *Network Security*, vol. 2018, no. 1, pp. 8–11, Jan. 2018. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1353485818300060

[14] T.-T. Kuo, H.-E. Kim, and L. Ohno-Machado, "Blockchain distributed ledger technologies for biomedical and health care applications," *Journal of the American Medical Informatics Association*, vol. 24, no. 6, pp. 1211–1220, Nov. 2017. [Online]. Available: https://doi.org/10.1093/jamia/ocx068

[15] R. R. O'Leary. (2018, Oct.) A solution to cryptos 51% attack? fine miners before it happens. Coindesk. [Online]. Available: https://www.coindesk.com/a-solution-to-cryptos-51-attack-fine-miners-before-it-happens

[16] B. S. Jyothi and D. Janakiram. Symon: Defending large structured p2p systems against a sybil attack. [Online]. Available: http://www.cs.kent.edu/~javed/class-FP2P10S/papers-2010/p02-SyMon_P2P09.pdf

[17] S. Nakamoto. (2008, Oct.) Bitcoin: A peer-to-peer electronic cash system. [Online]. Available: https://nakamotoinstitute.org/bitcoin/#selection-45.325-45.1124

[18] C. Prez-Sol, S. Delgado-Segura, G. Navarro-Arribas, and J. Herrera-Joancomarti. Double-spending prevention for bitcoin zero-confirmation transactions. [Online]. Available: https://eprint.iacr.org/2017/394.pdf

[19] S. Solat and M. Potop-Butucaru. (2016) Zeroblock: Preventing selfish mining in bitcoin. [Online]. Available: https://hal.archives-ouvertes.fr/hal-01310088v1/document

[20] R. Zhang and B. Preneel. Publish or perish: A backward-compatible defense against selfish mining in bitcoin. [Online]. Available: https://www.esat.kuleuven.be/cosic/publications/article-2746.pdf

[21] A. Singh, T.-W. Ngan, P. Druschel, and D. Wallach. Eclipse attacks on overlay networks: Threats and defenses. [Online]. Available: https://www.eecs.harvard.edu/~mema/courses/cs264/papers/eclipse-infocom06.pdf

[22] D. Cordell. (2015, Mar.) Colored coins, what they are and how they work on the bitcoin blockchain. [Online]. Available: https://bitcoinist.com/colored-coins-work-bitcoin-blockchain/

[23] P. Schueffel. Alternative distributed ledger technologies: Blockchain vs. tangle vs. hashgrapha high-level overview and comparison. [Online]. Available: https://www.researchgate.net/profile/Patrick_Schueffel/publication/324280397_Alternative_Distributed_Ledger_Technologies_Blockchain_vs_Tangle_vs_Hashgraph-A_high-level_overview_and_comparison/links/5ac9e481aca272abdc6158d5/Alternative-Distributed-Ledger-Technologies-Blockchain-vs-Tangle-vs-Hash.pdf

[24] M. Crosby, Nachiappan, P. Pattanayak, S. Verma, and V. Kalyanaraman. (2015, Oct.) Blockchain technology: Beyond bitcoin. [Online]. Available: https://scet.berkeley.edu/wp-content/uploads/BlockchainPaper.pdf

[25] Smart contracts. [Online]. Available: https://www.investopedia.com/terms/s/smart-contracts.asp

[26] J. Risberg. What is eos? everything you should know. [Online]. Available: https://coincentral.com/what-is-eos/

[27] J. V. Rossum. (2017, Nov.) Blockchains for research: Perspectives on a new paradigm for scholarly communication. [Online]. Available: https://www.digital-science.com/resources/digital-research-reports/blockchain-for-research/