**“Addressing Different Blockchain Implementations of Decentralized Finance.”**

**By**

**Eric Webb**

**ISEC660 - Advanced Network Security**

**Winter 2021**

**Professor : Dr .Wei Li**

**Table of Contents:**

**Abstract:……………………………………………………………………………………….....3**

**Key Words:………………………………………………………………………………………3**

**Introduction:…………………………………………………………………………………..3, 4**

**Introduction: Proof-of-Work VS Proof-of-Stake:………………………………………..4, 5, 6**

**Use Case Example 1 : Bitcoin (BTC) Proof-of-Work:…………………………6, 7 ,8, 9, 10 ,11**

**Use Case Example 2 : Monero (XMR) Proof-of-Work:………………………….11, 12, 13, 14**

**Use Case Example 3 : Tezos  (XTZ) Proof-of-Stake:…………………………………14, 15, 16**

**Conclusion:………………………………………………………………………………….16, 17**

**Future Work:…………………………………………………………………………………...17**

**References:.…………………………………………………………………………………..18,19**

**Abstract:**

This narrative will explore and argue the validity of the security aspects Decentralized Finance (DeFi) provides through popular DeFi protocols and their proprietary blockchain implementations. This was done by conducting research on popular DeFi protocols with each of them having a different blockchain implementation as their underlying architecture. Each one of these DeFi protocols provides security to transactions in their own independent way. The information provided for each individual use case includes validity to why decentralized finance transactions are secure through these DeFi protocols’ blockchain implementations, while also bringing to light the potential flaws that needed to be addressed. A comparison review was conducted between the DeFi protocols to better understand and grasp when and where to implement their blockchain architecture and leads to an argument on to why one would take precedence in certain use cases.

**Keywords:**

Blockchain, Cryptocurrency, DeFi, PoW, PoS, NFT, PKI, P2P , UTXO, FILO ASIC GPU CPU LRS OTPK, RingCT

**Introduction:**

The conception of blockchain has led to a renaissance of wealth in the global economy. Since the birth of Bitcoin and other cryptocurrencies, new means of creating value have emerged on the world market. At a high level, some of these examples to create wealth in this new world economy of cryptocurrency include but are not limited to: Transactions, Mining, Staking, Creating Non-Fungible Tokens(NFT’s), Creating Smart Contracts, Lending, and Yield-Farming. A cornucopia of these topics has led to the foundation of what is known as Decentralized Finance (DeFi). In this statement “DeFi refers to a financial system which relies for its security and integrity on distributed ledger technology.” (Gudgeon, L., Knottenbelt, W., Perez, D., and Werner, S. 2020) This is revolutionary because “Unlike regular finance where the identity of all participants is known, and correct behavior can be enforced via regulation, DeFi actors are pseudonymous and DeFi systems need other means to prevent users from misbehaving.” (Gudgeon, L., Knottenbelt, W., Perez, D., and Werner, S. 2020) This is done by “the absence of traditional credit-rating mechanisms, the system rules are typically enforced by incentivizing actors to behave according to the rules of the system.” (Gudgeon, L., Knottenbelt, W., Perez, D., and Werner, S. 2020) As you can see DeFi is an umbrella term that can encompass many topics of conversation. The main ones involving security aspects, such as how do these blockchain implementations establish trust between parties without relying on a centralized entity? And how can we be sure they are cryptographically secure?

**Introduction: Proof-of-Work VS Proof-of-Stake:**

Fundamentally these Defi protocols run off their own unique blockchain implementations. Although it should be noted that often time blockchain projects are open source and are cloned from one another, so that they might appear as similar, but are actually very meticulously specialized to perform a particular task. In Essence these blockchains should be “decentralized, democratic, and resilient. They use consensus-based replication to decentralize the ledger among many independent participants. Thus, it can operate completely decentralized and does not require trust in a single authority.” (Gupta, S., Hellings, S., and Sadoghi, M. 2021) Most infamously the first DeFi project to publicly come forth and do this was Bitcoin and its Proof -of-Work (PoW) consensus algorithm. “PoW is a well adopted consensus algorithm. In a PoW-based blockchain, blocks are added to the growing chain by miners who participate in mining.” (Ma, X., Zhang, S., and Zheng, K. 2020) This process of mining is where “miners repeatedly cryptographically hash the assembled block, each time with a different nonce. Whenever the hash from any miner falls below a target time t, whose maximum value is m, the corresponding block is mined, and the miner then has the right to add this block to the blockchain and publish it.” (Ma, X., Zhang, S., and Zheng, K. 2020) When A miner is granted access to publish the block, they are rewarded and that is their incentive to do so. It should be noted that the PoW consensus does have “several limitations, e.g., energy inefficiency, delay, and vulnerable to security threats.” (Dutkiewicz, E ., Hoang, T., Nguyen, C., Nguyen, N., Nguyen, T., and Niyato, H. 2019) . To overcome these hurdles, new consensus algorithms were developed to address specific issues. Most notoriously the Proof-of Stake (PoS) algorithm, which aims to achieve the consensus via proving the stake ownership. The aim of this is to reduce the computational requirements needed to support the network compared to that needed for the PoW algorithm. Proof of Stake works by “providing the user who has the highest stakes in each network with the opportunity to exploit. Having the highest stakes gives the miner credibility and assurance that he will not tamper with the ledger.” (Dorai, R., and Nair, P. 2021)The beauty of this is that that PoS consensus algorithm “consumes a lesser amount of energy when compared with Proof of Work Blockchain methodology.” (Dorai, R., and Nair, P. 2021) Essentially, PoS are allowed to perform transactions with less energy because assets are stored in an escrow in which the miner will lose if they are caught making fraudulent transactions. At a high-level a good example would be if you put $1000 into an escrow to gain the rights to solve and earn 1$ transactions. Typically, one would not lie about 1$ if they risk losing their own $1000. The same concept is applied with the proof of staking algorithm. You put money upfront in escrow to hedge against you making bad transactions. Generally, “Proof of Stake mechanisms have been proven as not scalable and hence more suitable for a private network setup and cannot be effectively adopted for large scale use cases.” (Dorai, R., and Nair, P. 2021)

All blockchain implementations derive from having a PoW or PoS consensus algorithm at their core. Similarly, to how all mechanized vehicles have an Automatic or Manual drive transmission. It is imperative the PoW and PoS algorithms be understood since they will be at the core architecture of the proceeding blockchain implementations. From the above context we have seen that it is widely accepted that PoW consensus algorithms are scalable but resource intensive, while PoS algorithms are generally less resource intensive but more difficult to scale. This creates a tit for tat in the blockchain ecosystem especially in the realm of security. Generally, when things are more resource intensive, they are quicker to burn out causing DOS. When it comes to scaling up on implementations not designed to scale, it leaves more vulnerability points unsafeguarded then that would be in a scalable application

Moving forward in this narrative we will begin to discuss popular real-world use cases of these blockchain implementations and discuss how they are established throughout DeFi. These topics will begin to define security aspects, and gain insight to these specific use case examples strengths and weaknesses.

**Use Case Example 1 : Bitcoin (BTC) Proof-of-Work**

It’s no secret that Bitcoin was the first of its kind, birthing the Proof-of-Work consensus algorithm. This made it become the first public DeFi asset. It was released by a pseudonymous author labeled Satoshi Nakomoto in 2009. BTC was architected it to be” a peer-to-peer (P2P) payment system employing a shared ledger called blockchain, where a consensus method is employed such that the system is not controlled by any central party.” (Dimaz, A., Dongxi, L., Liu, K., Steinfeld, R., Wijaya, J., and Yu, J. 2019) A security advantage of this is that the network acts as almost a hive mind reducing single points of failure since the distributed ledger is stored across many nodes immutably.

BTC and its PoW algorithm use a myriad of cryptographic techniques to establish security. To begin, it uses Public Key Infrastructure (PKI). The public key is an address on the Blockchain with valued tokens sent across the network recorded in that address. While the private key provides its owner access to their digitally held assets. (Dorai, R., and Nair, P. 2021) This P2P network provides a layer of anonymity since true identities are hidden behind public keys represented as a hashed value. That is how BTC provides security through confidentiality. This also provides a layer of security to your assets since you are in control of your keys and no single entity can take that from you. In contrast if one’s private keys were to get compromised, they would lose their security posture, being a single point of failure for the end user. A good reference would be you cannot hack a dollar bill in the physical world, but you can leave your wallet open to be stolen from.

Another security aspect of BTCs blockchain implementation is its ability to provide immutability through decentralization, this is done by having the community verify transactions and storing the hash values of transactions in Merkle root fashion into blocks that act as a double-ended queue (Deque) so data can be referenced up and down stream. (Dorai, R., and Nair, P. 2021) These Merkle root are a sum of all the transactions on the network stored as hashed values. This allows everyone to publicly download this chain of transactions and verify malicious actors. A real-world example of this would be if you checked your BTC wallet and saw your assets missing, you could check a block explorer to see where those assets were sent to from your wallet address and see what address now has them. This is how BTC’s blockchain offers security through availability because transactions are always visible on its distributed ledger.

These transactions are solved by independent nodes on the network called miners. These miners solve algorithms to be able to add blocks to the network and get rewarded for doing so. In a simple construct, this mining process can be defined as so : “miners repeatedly cryptographically hash the assembled block, each time with a different nonce. Whenever the hash from any miner falls below a target time t, whose maximum value is m, the corresponding block is mined, and the miner then has the right to add this block to the blockchain and publish it.” (Ma, X., Zhang, S., and Zheng, K. 2020) Bitcoin argues its decentralization by allowing anyone to participate as a miner node on the network. This is how it provides security through availability as decentralization, with the goal being that no single entity can take control of the network. An example of this would be if a government were to try and shut down BTC network in one country, then another participant in another country could support the network via mining. This provides resiliency and redundancy to support the network from a denial of service or stopping the transaction of funds because any node anywhere in the world can support the network and be rewarded for doing so.

As stated earlier there is a direct correlation between the target hash value and the number of miners on the network. The more miners on the network, the lower the target hash value conversely increasing the network difficulty. This is how BTC supports its decentralization through an elastic network difficulty. The beauty of this is that in theory one single entity with an exponentially stronger hashing power cannot overpower the network because they would be competing with themselves raising the network difficulty by lowering the target hash rate since their added hashing power is being detected by the network. This also supports BTCs security through availability because if a large set of nodes were to disappear, then other miners would have incentive to start again, since they could then mine the network with less difficulty. If A miner is lucky enough to produce a number that is smaller than the target hash value, then they are rewarded for adding a block to the chain. This reward is called a coinbase transaction and is agreed upon multiple nodes before being added to the ledger. This coinbase transaction is what creates new coins to be minted to the network and is the initiation point of reference of every coin and the wallet address it was created too. (Hohlfeld, O. , Rüth, J., Wolsing, K., and Zimmermann, T. 2018) The amount of coins to be distributed to rewarding miner depends on the amount of transactions on the network. BTC securely provides scarcity by the concept of an event called the “halvening”. Every so many transactions, the coinbase reward will half, so that the miner will only receive half of what they original were receiving. This is how BTCs blockchain guarantees the rate of which coins are minted will decrease as time goes by. Pairing with this is the fact the BTC will only mint 21 million coins, in the world of DeFi this argues the security and protection against inflation since more coins cannot be minted.

Another way BTCs blockchain instills security of its transactions is by employing an Unspent Transaction Output (UTXO) model for totaling users’ assets. This promotes security through integrity and availability. Traditionally in an account model type architecture all assets are computated and stored in a single repository. In UTXO, the account total is a sum of all the unspent assets from previous transactions. An Example is someone receives 1 BTC five times they blockchain would see 5 deposits of unspent transactions totaling 5 BTC. If that same user was to pay for an item that was 2 BTCs, the network would subtract 1 BTC from the first two transactions, operating in First in Last Out (FILO) order. Denominations can’t be broken up from the UTXO, so if change is needed the whole amount is sent and the receiving user sends what is owed. The beauty of this is that it supports security via Integrity and Availability of CIA triad by having more references of where the source of assets came from being publicly available.

From the dialogue above we can see how BTC’s PoW consensus algorithm implements some security aspects within its blockchain. This next section will begin to discuss some faults and present known vulnerabilities within BTCs blockchain implementation. A popular argument is that BTC’s PoW algorithm is resource intensive. This means it consumes a lot of energy and needs some well-manicured resources to operate when this effort could be better applied to other means of helping the world. The computational energy used to add a block to the chain is hard just for the sake of being hard. The computational energy does not need to be used as much in other blockchain implementations that apply security through other measures. Another known crucial issue is what is labeled the “51 percent attack”, in this vulnerability if 51% of the network agrees on a lie then that lie becomes the truth. This is very much theoretically possible because the top mining pools could collude together nefariously to influence the network in their favor. That is how the 51% attack effects BTCs security posture. This is due to the development of Application-specific Integrated Circuits (ASICs). Originally the network difficulty was so low that everyday home computers Central Processing Units(CPUs) could compete with each other to produce a coinbase transaction and add blocks to the network. Then Graphics Processing Units (GPUs) were discovered to be able to mine and create the BTC transaction taking a slight advantage over CPU mining. Then ASICs were developed, which are CPUs that are application specific to solving the BTC 256 happening embedded within the application. This has led to ASIC miners dominating the world market. If enough entities of ASIC miners came together, they could influence the market. (Hohlfeld, O. , Rüth, J., Wolsing, K., and Zimmermann, T. 2018) This is an apparent security flaw in the BTC network. Other blockchains have been developed to be ASIC resistant. The fact that some huge players have developed to supporting so much of the BTC network with their ASIC miners, it has argued the validity of BTCs decentralization. Another flaw regarding BTCs PoW algorithm is that of the UTXO architecture it deploys. Since UTXO has to add up sums of transactions to define a total it takes up more resources, not only do other properties need to be defined so data can be referenced but the data has to be compiled to a total sum, taking more time and processing power to sync. In contrast it is faster to have a total sum in a central account like in the account model. This adds to BTC’s slow transaction times because all that UTXO is being transmitted and processed behind the scenes.

From this we have established the basics of BTC’s blockchain algorithm as it was the first to market and the first to implement a PoW consensus algorithm using its UTXO. The above articles defined security aspects that both validate and argue the position of BTC security posture. Next, we will begin to define other security aspects of another PoW protocol.

**Use Case Example 2 : Monero (XMR) Proof-of-Work**

In the last section we learned about the basics of PoW through its conception via BTC blockchain application. This next narrative will expand upon that and begin to explore deeper into another PoW based blockchain called Monero (XMR). XMR is “a privacy preserving cryptocurrency whose PoW is designed to be GPU and ASIC resistant enabling CPUs to have an even playing field. Specifically, it uses the Cryptonite hash function in its PoW to mine a new block.” (Hohlfeld, O. , Rüth, J., Wolsing, K., and Zimmermann, T. 2018) Previously we have discussed how identities are presented as public keys, as a pseudonym for usernames. This is supposed to provide privacy to the BTC network. An issue that has arose is that “Researchers have developed analysis methods to reveal information about the users from external sources. The transparent blockchain data was also utilized to reveal the transaction patterns such that the users’ behaviors are identified.” (Dimaz, A., Dongxi, L., Liu, K., Steinfeld, R., Wijaya, J., and Yu, J. 2019) This has argued to make BTC less anonymous because the public key is more easily traceable since your transactions are public. Monero aims to solve that issue while still employing a PoW consensus algorithm.

MNR’s proprietary protocol is called the Cryptonite protocol. The beauty of this Cryptonite protocol is “the untraceability of the sender and the unlinkability of the receiver are protected by cryptographic methods such as Linkable Ring Signature (LRS) and one-time public key (OTPK) which makes transactions more anonymous.” (Dimaz, A., Dongxi, L., Liu, K., Steinfeld, R., Wijaya, J., and Yu, J. 2019) The LRS provides security from double spending by “combining several existing outputs (called decoys or mixins) which have the same amount of coins into a single input. These outputs must have a real output which will be spent in the transaction.” (Liu, D., Liu, J., Steinfield, R., and Wijaya, A. 2018) The purpose of this it so to reduce the possibility of an adversary to guess a real output over a given number of outputs. The kryptonite PoW consensus employs the OTPK so that observers who have access to the blockchain can not establish relationship between public keys and only the sender and receiver have access to those relationships. This is how XMR provides security through confidentiality with its anonymity. This is done by the “receiver sending a set of parent public keys to the sender. The sender then generates new child public keys by using random values, which are included in the transaction data in encrypted format. The receiver scans the network for new transactions and compute the secret key of each transaction by using the parent private keys she holds. If a result matches the destination key, then she includes the transaction in her wallet as an incoming transaction.” (Liu, D., Liu, J., Steinfield, R., and Wijaya, A. 2018) That is how public keys stay securely hidden to the outside world in XMR. As another means of anonymity, XMR also employs a technique labeled a Ring Confidentiality Transaction (RingCT) to hide the amount of coins in the transaction. This is employed in the LRS being the “main requirement for constructing a ring signature is that each ring member must have the exact same amount of coins and therefore the real one cannot be distinguished from the decoys.” (Liu, D., Liu, J., Steinfield, R., and Wijaya, A. 2018) This mitigates the problem of liquidity by hiding the amount of coins contained in the public keys also supporting security through confidentiality.

Now that we have defined how XMR provides added security through applied confidentiality, we will begin to define some know issues with this blockchains implementation. The first one being what is known as the Black Marbles Attack. This attack references black marbles to being false crafted mixin values, and white marbles being legitimate ones. The more black marbles in the Urn of marbles (Blockchain in this example) than the greater chance a black marble (Illegitimate mixin output value) will be chosen. (Dimaz, A., Dongxi, L., Liu, K., Steinfeld, R., Wijaya, J., and Yu, J. 2019) So essentially nefarious actors can craft redundant false mixin values for a better chance of getting chosen output. As stated, earlier XMR uses RingCT to guarantee mixins are applied in transactions as to not reveal the users public key. Prior to this update mixins were not guaranteed. It has been discovered that if a Zero-Mixing transaction was performed it can be traced. It is assumed that ”at least more than half of all analyzed inputs (prior to RingCT) can be distinguished between the decoys and the real outputs.” (Dimaz, A., Dongxi, L., Liu, K., Steinfeld, R., Wijaya, J., and Yu, J. 2019) This statement gives argument to XMR’s security through anonymity because it proves new ways are being determined to extinguish its anonymity.

Although anonymity features were implemented, research results show that the transactions in Monero can still be traced due to the problem of zero mixin transactions. (Dimaz, A., Dongxi, L., Liu, K., Steinfeld, R., Wijaya, J., and Yu, J. 2019) From this context we can see how XMR extended the capabilities of the PoW consensus algorithm to be more decentralized by being ASIC resistant and by providing security through confidentiality. From this we also gathered insight into known issues with XMRs anonymity with its previous bug the zero mixin and the black marbles attack for guessing outputs.

**Use Case Example 3 : Tezos  (XTZ) Proof-of-Stake**

In the previous section we discussed different PoW algorithms along with their respective strengths and weaknesses. In this section we will be leaving the PoW algorithm behind and start to focus on the PoS consensus algorithm. This use case example for PoS has been chosen to be Tezos (XTZ). At glance XTZ “is an innovative blockchain that improves on several aspects compared to more established blockchains. It offers an original proof-of-stake consensus algorithm and can be used as a decentralized smart contract platform. It has the capacity to amend its own economic protocol through a voting mechanism and focuses on formal methods to improve safety.” (Allombert, V., Bourgoin, M,. and Tesson, J. 2019)

From the above context we can see that XTZ uses its own proprietary PoS protocol, this is called the Liquid Proof of Stake (LPoS) consensus algorithm. To begin, one interesting way this LPoS provides security through availability is that it is a self-amending blockchain. This means that the protocol can be changed based on a super majority vote of %80 of the nodes. (Allombert, V., Bourgoin, M,. and Tesson, J. 2019) This provides security because the community is available to perform an on-chain voting consensus to support the network, this prevents from forking of blockchains when development teams have a dispute in how the blockchain architecture should be implemented.

Another way XTZ provides security to its blockchain by keeping it available is that small time owners can still represent votes on the network. XTZ allows holder of larger amount of coins to be able to delegate (produce or sign blocks) and be rewarded to do so. In XTZ they call these nodes a baker, when a small-time owner doesn’t have enough XTZ to bake, they can use their XTZ to be added to a baker’s liquidity. As stated, “participants that do not hold enough tokens or who do not wish to bake blocks can delegate their tokens to another baker, much like in Liquid Democracy. They keep the ownership of their tokens but increase the stake of their delegate in the random assignment of baking slots. Delegation makes the PoS system fairer and more participative and helps balance a possible concentration of tokens in few hands.” (Allombert, V., Bourgoin, M,. and Tesson, J. 2019) This is how XTZ provides security through availability because small time asset owners can still have a fair hand in the voting and staking rights of the network by providing their assets as liquidity to their delegates via it LPoS algorithm.

Although XTZ displays arguments to validate its security its LPoS is not with out fault. Because PoS consensus algorithms have a global predictability, attackers can use this information to perform a double spend attack. This effects XTZ because “participants in the consensus protocol are able to determine far in advance exactly when they will have the opportunity to mine blocks. This helps validators know when to create or validate a block, but this predictability also makes a double-spend attack easier to perform, as an attacker knows precisely when the opportunity to reorg will arise, allowing them to send a soon-to-be-deleted transaction to an unsuspecting counterparty at exactly the right time.” (Moroz, D., Neuder, M., Parkes, D., and Rao, R. 2020) This is how attackers can take advantage of the XTZ network with a double spend attack, by waiting for a reorg to delete a transaction and knowing precisely when to attempt to mine a block.

From this section we can see how XTZs LPoS consensus algorithm provides security through availability by making it an amendable chain and keeping it decentralized by allowing small time owners to have a voting and staking right. We also discussed how the LPoS is vulnerable to a double spend since transactions are broadcast ahead of time when they will be performed allowing for a vulnerability if a reorg is performed.

**Conclusion**:

From this narrative we have discussed the Pros and Cons of popular DeFi protocols and their respective PoW and PoS consensus algorithms, we have also discussed these implementations in real world use case examples. These use cases defined the security Pros of the DeFi protocols with BTCs blockchain implementation using UTXO to verify user account totals for integrity, Monero’s LRS that implements a RingCT and its OTKP for confidentiality, and XTZ’s decentralization from its LPoS and amendablity for availability. Besides from the security pros of these DeFi protocols, the vulnerabilities and weaknesses were also discussed. BTC being vulnerable to a 51% attack. Monero being vulnerable to a Black Marble attack. While XTZ being vulnerable to a double spend because of the global predictability factor.

It is in my conclusion that each DeFi protocol has its underlying use case and features specific to is purpose. If one were wanting a scalable solution that is not dependent on speed than a PoW algorithm would be preferred, if a smaller more private project that was dependent on speed was needed then a PoS algorithm would be recommended. BTC would be recommended as an everyday store of value, while XMR would be recommended for someone who wants their transactions to be more private in nature. XTZ would be recommended for an application that need the use of smart contract capabilities and would like a vote in the governance of the blockchains protocol.

**Future Work**:

Blockchain applications and their corresponding DeFi protocols will be the point of reference for my future work. In my future endeavors I will be dedicating time and resources to learn more about different types of consensus algorithms like Delegated Proof of Stake (DPoS) or Proof of Authority (PoA) to name a couple. More interestingly in my opinion I would like to publish my own blockchain application using some type of proprietary smart contract language such as Ethereum’s Solidity or XTZ’s Michelson. Lastly, I would like to perform more research into Non-Fungible Tokens (NFT’s) and publish my own.

**References:**

Allombert, V., Bourgoin, M,. and Tesson, J. (2019) "Introduction to the Tezos Blockchain," International Conference on High Performance Computing & Simulation (HPCS), Dublin, Ireland, 2019, pp. 1-10, doi: 10.1109/HPCS48598.2019.9188227.

Moroz, D., Neuder, M., Parkes, D., and Rao, R. (2020) Defending Against Malicious Reorgs in Tezos Proof-of-Stake. In Proceedings of the 2nd ACM Conference on Advances in Financial Technologies (AFT '20). Association for Computing Machinery, New York, NY, USA, 46–58. DOI:https://doi- org.ezproxylocal.library.nova.edu/10.1145/3419614.3423265

Dimaz, A., Dongxi, L., Liu, K., Steinfeld, R., Wijaya, J., and Yu, J.(2019). “On the Unforkability of Monero.” Proceedings of the 2019 ACM Asia Conference on Computer and Communications Security (Asia CCS '19). Association for Computing Machinery, New York, NY, USA, 621–632. DOI:https://doi- rg.ezproxylocal.library.nova.edu/10.1145/3321705.3329823

Dorai, R., and Nair, P. (2021) "Evaluation of Performance and Security of Proof of Work and Proof of Stake using Blockchain," Third International Conference on Intelligent Communication Technologies and Virtual Mobile Networks (ICICV), Tirunelveli, India, 2021, pp. 279-283, doi: 10.1109/ICICV50876.2021.9388487.

Gudgeon, L., Knottenbelt, W., Perez, D., & Werner, S. (2020). DeFi Protocols for Loanable Funds: Interest Rates, Liquidity and Market Efficiency. In Proceedings of the 2nd ACM Conference on Advances in Financial Technologies ( AFT '20 ). Association for Computing Machinery, New York, NY, USA, 92–112. DOI:https://doi- org.ezproxylocal.library.nova.edu/10.1145/3419614.3423254

Gupta, S., Hellings, S., and Sadoghi, M. (2021). Fault-Tolerant Distributed Transactions on Blockchain. Synthesis Lectures on Data Management. February, 2021. https://doi- org.ezproxylocal.library.nova.edu/10.2200/S01068ED1V01Y202012DTM065

Hohlfeld, O. , Rüth, J., Wolsing, K., and Zimmermann, T. (2018). “Digging into Browser-based Crypto Mining. In Proceedings of the Internet Measurement Conference 2018” (IMC '18). Association for Computing Machinery, New York, NY, USA, 70–76. DOI:https://doi-org.ezproxylocal.library.nova.edu/10.1145/3278532.3278539

Ma, X., Zhang, S., and Zheng, K. (2020) "Difficulty Prediction for Proof-of-Work Based Blockchains,"  IEEE 21st International Workshop on Signal Processing Advances in Wireless Communications (SPAWC), Atlanta, GA, USA, 2020, pp. 1-5, doi: 10.1109/SPAWC48557.2020.9154260.

Dutkiewicz, E ., Hoang, T., Nguyen, C., Nguyen, N., Nguyen, T., and Niyato, H. (2019) "Proof- of-Stake Consensus Mechanisms for Future Blockchain Networks: Fundamentals, Applications and Opportunities," in IEEE Access, vol. 7, pp. 85727-85745, 2019, doi: 10.1109/ACCESS.2019.2925010.

Liu, D., Liu, J., Steinfield, R., and Wijaya, A. (2018) "Monero Ring Attack: Recreating Zero Mixin Transaction Effect," 2018 17th IEEE International Conference On Trust, Security And Privacy In Computing And Communications/ 12th IEEE International Conference On Big Data Science And Engineering (TrustCom/BigDataSE), New York, NY, USA, 2018, pp. 1196-1201, doi: 10.1109/TrustCom/BigDataSE.2018.00165.