On Blockchain Security and Relevant Attacks

Joanna Moubarak
ESIB, USJ
CIMTI
Beirut, Lebanon
joanna.moubarak@net.usj.edu.lb

Eric Filiol
ESIEA
$(C+V)^O$ *Lab*
Laval, France
efiliol@netc.fr

Maroun Chamoun
ESIB, USJ
CIMTI
Beirut, Lebanon
maroun.chamoun@usj.edu.lb

***Abstract*—The blockchain technology witnessed a wide adoption and a swift growth in recent years. This ingenious distributed peer-to-peer design attracted several businesses and solicited several communities beyond the financial market. There are also multiple use cases built around its ecosystem. However, this backbone introduced a lot of speculation and has been criticized by several researchers. Moreover, the lack of legislations perceived a lot of attention. In this paper, we are concerned in analyzing blockchain networks and their development, focusing on their security challenges. We took a holistic approach to cover the involved mechanisms and the limitations of Bitcoin, Ethereum and Hyperledger networks. We expose also numerous possible attacks and assess some countermeasures to dissuade vulnerabilities on the network. For occasion, we simulated the majority and the re-entrancy attacks. The purpose of this paper is to evaluate Blockchain security summarizing its current state. Thoroughly showing threatening flaws, we are not concerned with favoring any particular blockchain network.**

***Index Terms*—Blockchain, Security, Bitcoin, Ethereum, Hyperledger, DLT, Attack.**

I. Introduction

Nowadays, all information systems, computational tasks and data storage operates in a distributed manner for reliability, accessibility, parallelism or geographical purposes. The data replication minimizes the latency and avert the loss. Moreover, several nodes may bear some lapses and continue to process functions properly. Nevertheless, distributed technologies have many advantages, they also present several problems. Typically, systems management remains a crucial challenge. Recently, blockchain unveiled the potential of DLTs (Distributed Ledger Technologies) and its applications. The study of fault-tolerant distributed systems and the validity of byzantine agreement started in 1970s [1]. Many researches included the CAP theorem [2] [3] and asynchronous models [4]. The blockchain technology was introduced later by Satoshi Nakamato [5] [6] in 2008 aiming to provide a solution for the double-spending[1] problem [7]. In 2013, Vitalik Buterin presented the Ethereum network [8] characterized by its ability to program the blockchain [9]. In 2014, Gavin Wood [10] implemented the Ethereum Virtual Machine (EVM). In 2015, the whisper protocol permitted secure messaging between nodes and sensitive Dapps (decentralized applications). In December 2017, the mobile first network infrastructure for ethereum was introduced [11].

[1]The capability of spending the same digital coins several times.

Primarily, blockchain is a peer-to-peer, consistent, fault tolerant distributed ledger network that utilizes cryptographic puzzles to achieve consensus and transactions agreement. This technology powered innovation and its utilization has spread across multiple domains and use cases beyond FinTech. On the one hand, smart contracts which are self-executing functions stored on the blockchain and utilized to create Dapps, are key-enablers for several businesses. Certainly, Dapps are decentralized and are preserved by the different peers in the blockchain network.

On the other hand, several systems flaws threaten DLTs security and privacy. This mainly depends on the consensus algorithm in use and its relative considerations. Our work, though comparing the three main DLTs in the market today, namely Bitcoin, Ethereum and Hyperledger, has the specificity to evaluate their security. The main purpose in our work is to spot the attention on several issues in each blockchain network. We covered scalability measures, consensus algorithms, mining computations and nodes fairness. Particularly, we tackle security and privacy of each blockchain network. First, the analysis couples the several DLTs exposing their common characteristics. Furthermore, our comparison, allows identifications of flaws which lead to a higher probability of attack. Moreover, we conducted a majority attack targeting the bitcoin network and performed a re-entrancy attack on the Ethereum network.

This paper is structured as follows. Section II details and compares the blockchain networks mechanisms. Section III examines the security challenges of blockchains networks and Section IV exposes multiple possible attacks on the blockchain. Finally, Section V concludes the paper and states our future work.

II. Comparison of Blockchain Platforms

In this section, we overview the key concepts of Bitcoin, Ethereum and Hyperledger architectures. We detailed this part in [12].

A. Blockchain Overview

The blockchain is a peer-to-peer, fault-tolerant network of nodes, working on the principle of transactions agreement and operating without any central entity. For this intent, all parties mutually validate transactions following an agreed consensus algorithm and aggregate them into blocks, constructing the

978-1-5386-1254-5/18/$31.00 ©2018 IEEE

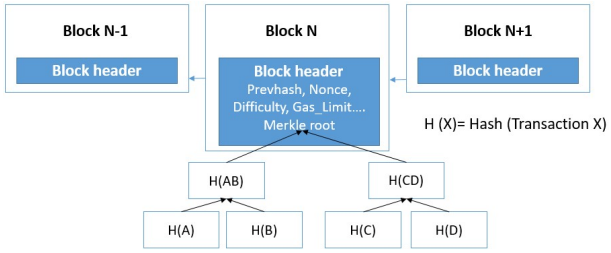

Fig. 1. Blockchain structure

structure of the blockchain. Transactions signing and verification mechanisms utilize cryptographic algorithms, precisely Elliptical Curve Digital Signature Algorithm (ECDSA), to establish the relations between the several blocks. In order to confirm entries, a merkle tree root hash is used to condense the data hashed into blocks. Each header holds the value of a previous hash, a nonce, the charge of a transaction (gas), a mathematical measure to assess the accuracy of a transaction (difficulty) and other constraints that diverge between DLTs (see Fig. 1). For instance, in an Ethereum header, three trees are available for transactions outcomes and account handling [13]. Also, the hyperledger block has a distinct structure alienated between block header, block data and block metadata [14].

B. Features

Regardless of the DLT type, the blockchain technology offers numerous security characteristics:

- Immutability: Once added to a block, a transaction becomes irremovable.
- Auditability: Each block is characterized by cryptographic schemes and secure timestamping offering the capacity to audit each transaction.
- Integrity: The SIGHASH function validates the signatures ensuring that any modification will invalidate the transaction.
- Authorization: Elliptical Curve Digital Signature Algorithm (ECDSA) is used to create the links between the blocks.
- Fault Tolerance: Many agreement mechanisms are involved to achieve the consensus in DLTs.
- Transparency: The transactions are appended into blocks and replicated publicly to the peers.
- Availability: Even if peers exit the network, the blockchain network is continually available.
- Consistency: Once the miners agree on the consensus and block arrangement, the distributed ledger is consistent and changes are infeasible.
- Privacy: While the distributed ledger is public, keys relatives to each parties are anonymous.

A detailed comparison between the several DLTs is abridged in Table I.

C. Blockchain networks

The Bitcoin network is a distributed network where digitally signed transactions are maintained by a public ledger representing the same data. In order to interrelate with the blockchain, a transaction is broadcasted into the network and validated by all peers. A block holds the information related to each transaction and once a consensus is achieved, the block will be appended the blockchain. The process will start over for every new transaction.

As for Ethereum and Hyperledger, the same set of interactions is used. However, a new concept of smart contract is introduced. Each contract instance will be wrapped up in the payload of each transaction and will circulate in the network depending on the gas limit, etc. Each receiving peer will execute smart contract functions creating a native replica of each contract in their current state.

D. Permission-less or Permissioned Network

Distributed Ledger Technologies can be alienated between permission-less and permissioned networks. A permissioned network limits the number of peers who can access the blockchain and participate in the validation in contrary to a permission-less network where everyone can contribute in the canonical chain. For instance, Bitcoin and Ethereum are permission-less blockchains that rely on a a Proof-of-Work[2] (PoW) consensus. This later is based on the machine power to resolve the cryptographic challenges consuming large amounts of electricity [15]. Therefore, the resolution and reward probabilities are higher for nodes with powerful hardware leading to regulate the network. To mitigate this problem, the new Ethereum Serenity release is moving to the Casper [15] protocol which is a Proof-of-Stake (PoS) algorithm where the different nodes are penalized if they bet incorrectly on a block. Also, one more benefit of this type of consensus is that the PoS is a less time consuming algorithm.

On the other hand, Hyperledger is permissioned blockchain. The number of participants is limited and managed by the system. The distributed ledger only include a set of definite transactions associated to each node. In a Hyperledger DLT, when a node broadcast a transaction, it will be forwarded to the ordering service to apply numbering identification in order to the peers to agree later on the same transactions sequence [16].

E. Scalability

Scalability remains a main challenge in DLTs, specifically for the Bitcoin and Ethereum networks where a copy of the complete history needs to be stored in each node. As for the permissioned hyperledger technology, this network can scale independently for each node without any disruption [12] because peers are abridged into endorsers, committers and consenters. Also, the parallel transactions processing in the hyperledger blockchain leads to a higher throughput.

[2]An algorithm which is hard to produce but easy to confirm and which contents certain rules based on computer computation.

TABLE I
DLTs COMPARISON

Features	Bitcoin	Ethereum	Hyperledger
Community	Bitcoin developers	Ethereum developers	Linux Foundation
DLT type	Permission-less	Permission-less	Permissioned
Currency	BTC	Ether	None
Consensus	PoW (based on SHA-256)	PoW (Ethash)	PBFT (excluding Corda)
Nb of nodes	Open	Open	Agreed
Actors	Peers	Peers	Peers, Orderers
Private transaction mode	No	No	Yes
Stimulus	Economics incentive, fees and rewards	Economics incentive, fees and rewards	Reputational Risk
Censorship resistance	No	Yes	No
Limit	7 transactions/sec	20 transactions/sec	No
State concept	No	Data	Key-value
Cross-contracts	No	Yes	Yes
Scalability	No	No	Yes
Block time	10 min	15 secs	Subject to the peers involved
Auditing Mechanism	No	No	Yes
Anonymity Mechanisms	No	Yes (with the whisper protocol)	Yes
Routing actors	No	Whisper mechanisms	Validators, auditors
GPU cost	Yes	Yes	No
Content Address Networks	Node hash	Swarm	Industry focused distribution
Client direct connections	No	Yes (through the Whisper protocol)	Yes
Applications	Digital Registry, Crypto Currency	Digital Registry, Crypto Currency, Smart Contracts	Digital Registry, Smart Contracts
Smart contract languages	No	Solidity, Serpent, Mutan, LLL	Chaincode
Languages	C++	Golang, C++, Python	GoLang, Java
Variants	+ 700 variants	Olympic, Frontier, Homestead, Metropolis (Future release), and Serenity (To be announced)	Burrow, Fabric, Iroha, Corda, Sawtooth

Principally, the one-megabyte block size in a bitcoin network not only outcomes delays (7 transactions/second) but also leads to the drop of not conform blocks. To overcome this limitation a Request Management System based on advertising requests was introduced as well as static time-outs. Moreover, in [17], bidirectional channels were presented. Also, in Ethereum, many functions are duplicated leading to scalability issues. Thus, the introduction of state channels and plasma channels in Ethereum networks to perform some transactions offline and conduct off-chain activities which leads also to enforce anonymity. Furthermore, in [18] , the sharding techniques which consist in splitting accounts states into separated chunks, were introduced.

III. DLTs SECURITY CHALLENGES

The main concept in blockchains is decentralization and it has imperatives allegations on the security. Blockchain mining malware and DoS attacks are reported constantly. In this section, we analyze smart contracts and lightweight clients vulnerabilities in addition to networks and consensus challenges in Blockchains.

A. Consensus

Several consensus algorithms [12] have been employed in DLTs depending on the altcoin type. For Bitcoin and Ethereum early releases the PoW algorithm suffers from multiple drawbacks. First, based on Moore's law, computation power will double every two years and the block difficulty will grow exponentially over time (see Fig. 2) thus a malicious miner can assemble hashing power of authentic validators and forge an attack at time T2 $>$ T1 to model the entire history at time T1 [19]. Besides, if the network dramatically looses mining computation power, block creation decelerates widely. Thus, the necessity to increase block difficulty exponentially and the need to provide enough soft fork time for all parties to updates their states. Regarding the Ethereum's alteration to a PoS algorithm for a better consensus in the new release, a new attack vector came upon this later. The External Reward Attack (ERA) presented in [20] aims, from a particular reward value, to increase steakers affluence over time even if the bet hasn't changed, allowing them to gain money 33% faster. As for the Practical Byzantine Fault Tolerant (PBFT) consensus utilized in some hyperledger networks, it relies on machine state replications and is capable of arbitrary toleration [21]. Therefore, it depends on stateless application only. Furthermore, applications relying on a strong cryptography are not reinforced allowing opponent access to some replicas [21].

B. Wallets

The principal purpose of wallets is to keep private keys. Several types have been utilized namely paper wallets, mobile wallets, desktop wallets, hardware wallets and online wallets. Preserving private keys from loss or theft is crucial. However, these wallets are subjects to failure or attacks, and in some scenario not irreversible. The introduction of multi-signature addresses and cloud storage systems may enable damage resistance unless collusion.

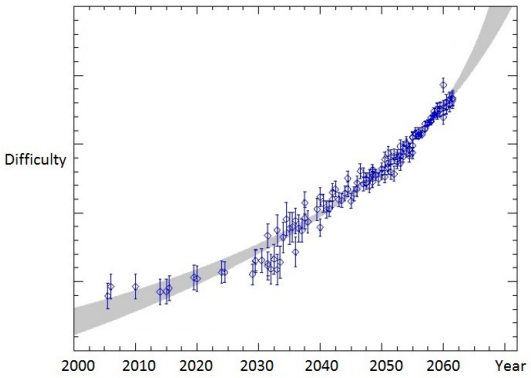

Fig. 2. Moore's Law

C. Smart Contracts vulnerabilities

Mostly all smart contracts are prone to faults due to the programming language weaknesses, especially in an Ethereum network.
Besides, after a block is appended to a chain, code errors are inept to be removed because transactions are permanents. As a solution, the implementation of auditing mechanism that involves the checking and reviewing of smart contract functions preserves a few of their flaws. Also, another approach consists in writing smart contract with an induced validity date after which this later will expire. Moreover, publicly exposing new vulnerabilities will let the attackers perform duplicitous actions. Furthermore, during the time of an attack, many DLT functions become unuseful. For instance, when deploying a transaction with an infinite loop, as long as gas fees are sustained, the operation is valid but not effective. Also, Ether may be lost in the transfer if the address of the recipient is not valid. Many solutions for these issues have been recognized in a Hyperledger network, introducing smart contracts endorsement to eliminate security flaws.

D. Lightweight clients

Inevitably, the security included in lightweight clients is lower than in the standard nodes. These nodes are light and simplified versions of nodes that contains only chunks of the blockchain. However, this leads to the loss of important information such as peers addresses and chain height. Therefore, malicious nodes may forbid the adoption of the chain forged by the light client by convincing this later that his chain is not the longest chain. Moreover, owning several bloom filters deceases the privacy. Besides, in order to reduce the bandwidth, light client take advantage of bloom filters that may result in false positives. Moreover, an adversary can make a link between the filters and the client to learn its address. To encounter this issue, using anonymization system like Tor might be advantageous.

E. Cryptography contravention

ECDSA [22] and secp256k1 [23] are considered strong cryptographic schemes. However, they might be cracked by private key harvesters in the far future. Thus, an attentive and innovative approach needs to be considered in future releases architectures.

IV. Attacks

This section exposes multiple attacks scenarios that can be performed on DLTs and summarizes the results of the two attacks that we have simulated.

A. Transactions Security

Blockchain security depends on cryptographic schemes and is based on keys management over the network. The transactions authenticity is corroborated applying digital signatures and each transaction point to the previous one. Each transaction is broadcasted between the peers for validation. Whereas, this allows adversary to delay the message delivery and may help in conducting double-spending. Also, another implication is denying the delivery of transactions [24]. Moreover, a more recurrent example of exploitation consists in controlling the target user inbound and outbound connexions by implementing an eclipse attack [25]. In [26], a double-spending attack was performed during block forks where typically in similar scenario the longest chain is adopted.

B. Spam attacks

A spam attack consists in pledging transactions that bear how users handle data, decelerating the network and delaying the creation of blocks while loosing gas and computation power. This result in decreasing of the number of reachable peers and entire network outage [27].

C. Malicious Contracts

Smarts contracts cannot handle code exceptions and procedures restructuring while transactions are validated. Transactions may become postponed or inaccurate [28]. For instance, we simulated a Re-entrancy attack [29] using Ganache [30] that provides a web interface for the truffle framework, installs on our machine a private blockchain network of five nodes and interacts with our Ethereum wallet where our malicious smart contract is created and signed. The purpose is to execute the same reentrant function by a contract before the initial process has terminated. Using the call function will invoke the interaction with the main contract several times before its execution is completed, causing bugs [31]. As shown in Fig. 3 and Fig. 4, there has been contract creation without spending Ether (VALUE=0.00 ETH) whereas the user has lost 0.01 ETH (BALANCE=99.99 ETH). A specific gas amount is utilized in our PoC. However, if the attacker called back multiple times without any gas limitations, the execution will continue until the user looses all his funds or until the maximum call stack depth is reached.

D. Anonymity

Metadata exposure disturbs the participants anonymity and reduce confidentiality. Besides, being aware of a user's address will lead to the history and exchanges tracking for financial and non-financials applications. Thus, the implementation of

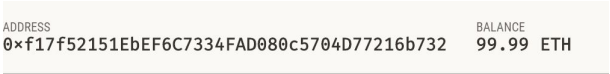

Fig. 3. Re-entrancy transaction

Fig. 4. Re-entrancy problem

pseudo-anonymity identity mixer relying on cryptographic schemes in the Hyperledger network and the introduction of the Whisper routing protocol in Ethereum blockchain to conceal sensitive interactions. However, a malevolent mixer can misbehave. Furthermore, in order to achieve better privacy, the concept of channels was presented. The idea is to segregate the activities and do not use the blockchain for every transaction. Off-chains functions that were originally conceived for scalability purposes have advantages on users secrecies as well. Moreover, the hyperledger network or other systems create private chains or consortium of chains, which, if performed properly characterize a state channel and if disadequately configured will lead to many privacy and security issues. Recently, ZK-SNARKs mechanisms were introduced to offer more processes integrity and selective transparency without compromising the privacy [32]. Whereas, for some applications, it takes 1 hour to forge a proof.

E. Mining pools

Joining a mining pool needs to be performed using consciousness. The control over a mining consortium depends on the hashing rates and the number of confirmations. In a PoW consensus, these parameters cannot mitigate the risk of attack. For instance, we simulated a majority attack targeting the bitcoin network. This attack permits to take control over the mining hashing power and thus conducting other malevolent behaviors. For that purpose, we connected to test network of ten different nodes. The results of our simulation are abridged in Table II. We demonstrated that a lower amount of confirmations will lead to a higher risk level.

Probability (0.4, 3)= 0.664168.
Probability (0.4, 4)= 0.603401.

Also, as we have experimented, owning 50% of the hashing power leads to take control over the network and obtaining the longest chain of transactions faster than the rest of the network.

Probability (0.5, Nb of confirmations)= 1.

TABLE II
MAJORITY ATTACK SIMULATION

Hashing Power Rate	Nb of confirmations	Probability of Attack
0.3	5	0.177352
0.3	6	0.132111
0.4	**3**	**0.664168**
0.4	**4**	**0.603401**
0.4	6	0.50398
0.5	2	**1**
0.5	3	**1**

This type of attack is a consequence of other malevolent scenarios such as stealing user machine power through trojan system exploitation [33], use timestamps dependency attack to manipulate the mining or add Call stack depth limit function and loops that can trigger gas limit.

F. Targeted DDoS attacks

These attacks consist in flooding the network by lots of information in a way it become irresponsive to transactions [34]. Besides, taking main peers offline for some period of time in order to forge additional validation results in serious consequences in DLTs applications. These latter theoretically include several mechanisms to protect against attacks on resiliency and availability. We can cite client protections related to the amount of check-signature, bloc size limitation, orphan transactions banning and ignorance of not standard transactions.

G. Timejacking attacks

An attacker could typically change node's network time counter by involving as multiple peers as possible and broadcasting wrong timestamps in the network. This results in speeding up other peers and isolating the target out of the network without interference from authentic nodes [35]. The tightening of time range and the usage of peer's system timing or Blockchain median time, might be possible solutions.

V. Conclusion

In this paper, we evaluated the security of blockchains specifically Bitcoin, Ethereum and Hyperledger networks. Moreover, we overview several DLTs challenges and attacks scenarios. Furthermore, we conducted a majority attack simulation visualizing the risk of a PoW consensus. Besides, we exploited the solidity language by performing a re-entrancy attack.

The aforementioned vulnerabilities present huge risks on blockchains networks leading to serious consequences. Possible countermeasures consist in using a recent timestamped address and utilizing it once, inflicting penalties, connections filtering, adding time-outs to queries and multiple auditing mechanisms.

To conclude, securing DLTs rely on the capacity to protect the private keys. Nevertheless the Blockchain advancement during the late ten years, many security issues and scalability challenges necessitate to be tackled.

Finally, businesses must pay attention when designing their applications and several characteristics are to be considered while choosing the best model to use. For instance, legal and regulatory frameworks are needed to manage blockchains and their uses. Thus, our next step consists in exploring blockchain malicious applications.

References

[1] R. Wattenhofer, *Distributed Ledger Technology - The science of Blockchain*. Forest Publishing, 2017.

[2] E. A. Brewer, "Towards robust distributed systems," in *PODC*, vol. 7, 2000.

[3] A. Fox and E. A. Brewer, "Harvest, yield, and scalable tolerant systems," in *Hot Topics in Operating Systems, 1999. Proceedings of the Seventh Workshop on*. IEEE, 1999, pp. 174–178.

[4] S. Gilbert and N. Lynch, "Brewer's conjecture and the feasibility of consistent, available, partition-tolerant web services," *Acm Sigact News*, vol. 33, no. 2, pp. 51–59, 2002.

[5] Coindesk, "Bitcoin technology," https://www.coindesk.com/bitcoin-technology-anonymoustor-network-more-powerful/.

[6] "History of bitcoin," http://historyofbitcoin.org/, 2008.

[7] S. Nakamoto, "Bitcoin: A peer-to-peer electronic cash system," 2008.

[8] V. Buterin *et al.*, "A next-generation smart contract and decentralized application platform," *white paper*, 2014.

[9] H. Diedrich, "Ethereum: Blockchains, digital assets, smart contracts, decentralized autonomous organizations," 2016.

[10] G. Wood, "Ethereum: A secure decentralised generalised transaction ledger," *Ethereum Project Yellow Paper*, vol. 151, 2014.

[11] G. Network, "From the winning team at ethwaterloo world's largest ethereum hackathon," https://gonetwork.co/, 2017.

[12] J. Moubarak, E. Filiol, and M. Chamoun, "Comparative analysis of blockchain technologies and tor network: Two faces of the same reality?" in *Cyber Security in Networking Conference (CSNet), 2017 1st*. IEEE, 2017, pp. 1–9.

[13] C. Dannen, "Introducing ethereum and solidity."

[14] "Hyperledger fabric blog," http://blockchain-fabric.blogspot.com/2017/04/hyperledgerfabric-v10-block-structure.html.

[15] A. Bahga and V. Madisetti, "Blockchain applications: A hands-on approach," 2017.

[16] "Hyperledger blog," http://hyperledger-fabric.readthedocs.io/en/latest/arch-deepdive.html.

[17] C. Decker and R. Wattenhofer, "A fast and scalable payment network with bitcoin duplex micropayment channels," in *Symposium on Self-Stabilizing Systems*. Springer, 2015, pp. 3–18.

[18] M. Scherer, "Performance and scalability of blockchain networks and smart contracts," 2017.

[19] S. Barber, X. Boyen, E. Shi, and E. Uzun, "Bitter to better how to make bitcoin a better currency," in *International Conference on Financial Cryptography and Data Security*. Springer, 2012, pp. 399–414.

[20] M. Uddin, "Attacking casper at ethereal hackathon," 2017.

[21] N. Chondros, K. Kokordelis, and M. Roussopoulos, "On the practicality of practical byzantine fault tolerance," in *ACM/IFIP/USENIX International Conference on Distributed Systems Platforms and Open Distributed Processing*. Springer, 2012, pp. 436–455.

[22] E. Rykwalder, "The math behind bitcoin," 2014.

[23] H. Mayer, "Ecdsa security in bitcoin and ethereum: a research survey," 2016.

[24] A. Gervais, H. Ritzdorf, G. O. Karame, and S. Capkun, "Tampering with the delivery of blocks and transactions in bitcoin," in *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security*. ACM, 2015, pp. 692–705.

[25] E. Heilman, A. Kendler, A. Zohar, and S. Goldberg, "Eclipse attacks on bitcoin's peer-to-peer network." in *USENIX Security Symposium*, 2015, pp. 129–144.

[26] A. Gervais, H. Ritzdorf, and G. O. Karame, "Double-spending fast payments in bitcoin due to client versions 0.8. 1," 2013.

[27] L. Parker, "Bitcoin spam attack stressed network for at least 18 months, claims software developer," https://securitycommunity.tcs.com/infosecsoapbox/articles/2017/11/04/bitcoin-under-attack, 2017.

[28] N. Atzei, M. Bartoletti, and T. Cimoli, "A survey of attacks on ethereum smart contracts (sok)," in *International Conference on Principles of Security and Trust*. Springer, 2017, pp. 164–186.

[29] Github, https://github.com/ethereum/wiki/wiki/Safety, 2016.

[30] "Working with ganache," http://truffleframework.com/docs/ganache/using.

[31] D. Wong, "Attacks on ethereum smart contracts," https://www.cryptologie.net/article/423/attacks-on-ethereum-smart-contracts/, 2017.

[32] V. Butterin, "Zk-snarks: Under the hood," https://medium.com/@VitalikButerin/zk-snarks-under-the-hood-b33151a013f6, year=2017.

[33] K. Huang, "Security 101: The impact of cryptocurrency-mining malware," https://www.trendmicro.com/vinfo/us/security/news/cybercrime-and-digital-threats/security-101-the-impact-of-cryptocurrency-mining-malware, 2017.

[34] R. R. O'Leary, "Bitcoin gold website down following ddos attack," https://www.coindesk.com/bitcoin-gold-website-following-massive-ddos-attack/, 2017.

[35] C. S. Community, "Bitcoin under attack!!" https://securitycommunity.tcs.com/infosecsoapbox/articles/2017/11/04/bitcoin-under-attack, 2017.