Future Generation Computer Systems 113 (2020) 183–195

Contents lists available at ScienceDirect

Future Generation Computer Systems

journal homepage: www.elsevier.com/locate/fgcs

On distributed ledgers security and illegal uses

Joanna Moubarak [a,*], Maroun Chamoun [a], Eric Filiol [b,c]

[a] Faculty of Engineering, USJ, Beirut, Lebanon
[b] Department of Cybersecurity, ENSIBS, Vannes, France
[c] National Research University, HSE, Moscow, Russia

ARTICLE INFO

Article history:
Received 31 December 2017
Received in revised form 15 September 2019
Accepted 24 June 2020
Available online 1 July 2020

MSC:
00-01
99-00

Keywords:
Blockchain
Bitcoin
Ethereum
Hyperledger
Algorand
Ripple
IOTA
Tangle
Smart contracts
Hashing
Security
Cryptography
Trust
Attacks
Vulnerabilities
Consensus

ABSTRACT

Distributed ledgers stimulate innovative services and enabled new applications in several domains, creating new concepts for trust and regulation. However, this backbone that is enabling novelties and abridging businesses comes with drawbacks and security flaws. In this paper, we evaluate several Distributed Ledger Technologies (DLTs) features depicting the Bitcoin, Ripple, Ethereum, Hyperledger, Algorand and IOTA networks. We focus on their security challenges and expose numerous threats and vulnerabilities. For instance, we have simulated a few of their possible attacks proving them non-immune. In the other hand, we show a few of their malicious use cases. Meticulously presenting DLTs menaces and flaws, we are not involved in preferring any specific DLT network.

© 2020 Elsevier B.V. All rights reserved.

1. Introduction

The science behind distributed ledgers started before cryptocurrencies appeared [1]. Several studies and algorithms emerged in order to provide fault-tolerant distributed systems using signatures to authenticate messages and limit byzantine nodes. State replication and randomized consensus protocols in synchronous and asynchronous systems provided the basics of DLTs. Digital signatures and crypto-based protocols secured the nodes communications. Also, quorum systems modeled the load and the dynamicity of peers joining the networks. Furthermore, the Consistency, Availability and Partition-Tolerance (CAP) theorem was introduced to mitigate distributed systems challenges, hence, protecting the exchange of information and the state of networks. Moreover, transactions and business logics that are matching nodes' inputs to specific outputs, linked the data structures that are shared among nodes.

Many facts ensured the development of DLTs. These architectures attracted several communities [2] and fueled innovations in different fields [3]. Also, the digitization of assets has spread across many applications [4]. DLTs are providing anonymity inside the systems [5] [6] and their market is making a great influence [7]. On the other hand, attackers are attempting to hack exchanges [8] and use this double-edge technology not only for scientific development but for illegal use cases as well [9–11]. Nevertheless, various challenges still ahead of their evolution, and many disadvantages need to be considered when designing new services. This paper goes beyond the study of DLTs, comparing them and analyzing their flaws. We question nodes equity, anonymity, privacy, fairness and transparency in the network. We summarize the features of Bitcoin, Ripple, Ethereum, Hyperledger, Algorand and IOTA, exposing the technologies behind

* Corresponding author.
E-mail addresses: joanna.moubarak1@usj.edu.lb (J. Moubarak), maroun.chamoun@usj.edu.lb (M. Chamoun), eric.filiol@univ-ubs.fr, efiliol@hse.ru (E. Filiol).

https://doi.org/10.1016/j.future.2020.06.044
0167-739X/© 2020 Elsevier B.V. All rights reserved.

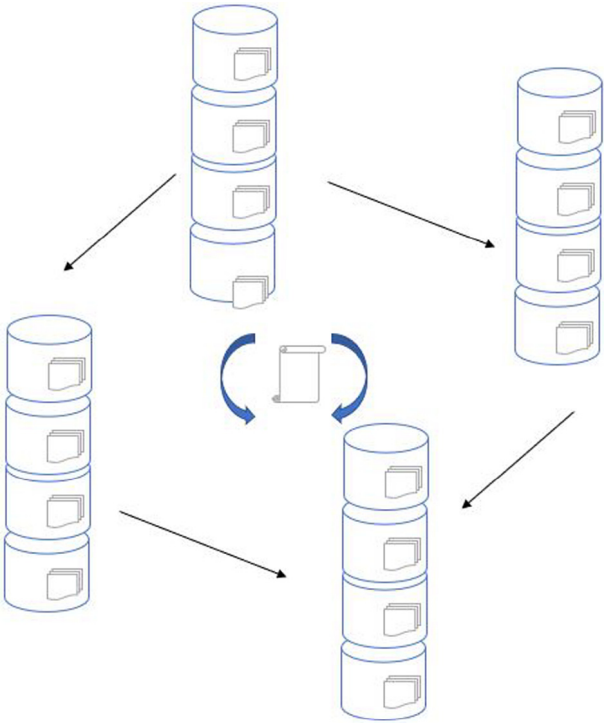

Fig. 1. Distributed ledgers.

these systems focusing on their vulnerabilities and security challenges. For instance, we describe possible attacks against DLTs providing some countermeasures to deter them. Furthermore, we simulated a few of these attacks proving the drawbacks exposed. Definitely, we are not concerned with favoring a technology among the systems analyzed. However, we aim to spot attention on multiple security concerns that need to be resolved in the next-generations DLTs.

This paper is organized as follows. Section 2 overviews the evolution of distributed ledgers. Section 3 compares Bitcoin, Ripple, Ethereum, Hyperledger, Algorand and IOTA DLTs. Section 4 exposes their security challenges and Section 5 shows their relevant attacks. We conclude this paper in Section 6.

2. DLTs development

The distributed ledger technology is a peer-to-peer network providing a tamper-evident, shared digital ledger where transactions are recorded (Fig. 1). The basis of the DLTs were described originally by L. Lamport with the Paxos algorithm [12,13]. In 1999 and 2000, E. Brewer and A. Fox surveyed the CAP theorem [14,15]. Also, in 2002, S. Gilbert and N. Lynch suggested an extension to asynchronous models [16]. Moreover, scientific literature deals with fault tolerant distributed systems as early as the late 1970s [1]. Basically, DLTs entail two main categories: the blockchain technology and the blockchainless DAG. This section is a brief overview of the development of distributed ledgers [17].

2.1. Bitcoin

The blockchain technology was elaborated by Satoshi Nakamato, which is speculated to be a codename for a collective of people in 2008 in the whitepaper entitled "Bitcoin: A Peer-to-Peer Electronic Cash System" [18] that describes the framework and introduces a solution for the double-spending[1] problem focusing on the main requirements to publicize all transactions and the ability to agree on their sequence. Fig. 2 illustrates Bitcoin core design. Before being appended to a block, a transaction is stored in the Mempool and stored on the RAM of the node. Each miner, using computer resources, tries to validate the block solving the crypto challenge and then get rewarded. Blocks and forks are clarified in [19]. The validation engine relies mainly on the wallet and headers to generate the result. Besides, an external application can interact with the blockchain network depending on the Bitcoin RPC that is used for authentication. Bitcoin framework marked a turning point and several communities offered numerous improvements to the Bitcoin protocol and proposed alternatives types of coins (alt-coins) each relying on their own network [4].

2.2. Ripple

In 2012, in order to address Bitcoin flaws, the Ripple Protocol was developed to directly transfer money between two parties [20]. The basic principle is that the transactions which are verified by the majority of the nodes are taken into consideration. The Ripple network, merely controlled by Ripple Labs, facilitates the exchange between several currencies. Nowadays, many development projects are based on Ripple. This DLT has been widely adopted by international money transfers and payment providers [21,22].

2.3. Ethereum

In 2013, Vitalik Buterin introduced Ethereum [23] which is an open stateful programmable blockchain [24] that allows more functionalities due to its architecture based on the concept of accounts, smart contracts utilizing turing-complete scripting language and actual balance outlaying Ether while in the first generation, Bitcoin relied on the concept of unspent account value. However, Ether will not become a broadly adopted digital currency, since from Ethereum 2.0, smart contract might be utilizing Bitcoin or Zcash [24].[2] In 2014, Gavin Wood [25] formally defined the Ethereum Virtual Machine (EVM) and implemented later the Ethereum chain using solidity as programming language for smart contracts. In 2015, in order to provide effective pure communications between decentralized applications (Dapps), where consensus on the blockchain is not required (see Fig. 3), the whisper protocol was designed [26]. This later is a combination of a hybrid distributed hash table (DHT) and a messaging system where messages with lower Time to Live (TTLs) are prioritized. Whisper operates in the dark mode, by means only minimal routing information is revealed using the probabilistic message forwarding. Besides, many functions in an Ethereum network need to be replicated between the users causing scalability concerns. In May 2016, a group of researchers announced swarm [27] which is a decentralized storage (see Fig. 3) and a content distribution service to be built on the Ethereum blockchain aiming to let the users/nodes having an excess of storage space or bandwidth to share it by incentivizing the operations.

Later in 2016, Vitalik Buterin presented "Ethereum 2.0 Mauve Paper" that concentrated on achieving the efficiency of the proof-of-stake[3] (PoS), the fast block time, the economic finality, the

[1] The ability to spend twice the same token in a digital currencies system.

[2] New cryptocurrency that focuses on the privacy within the blockchain network.

[3] A blockchain algorithm aiming to achieve consensus within the distributed network.

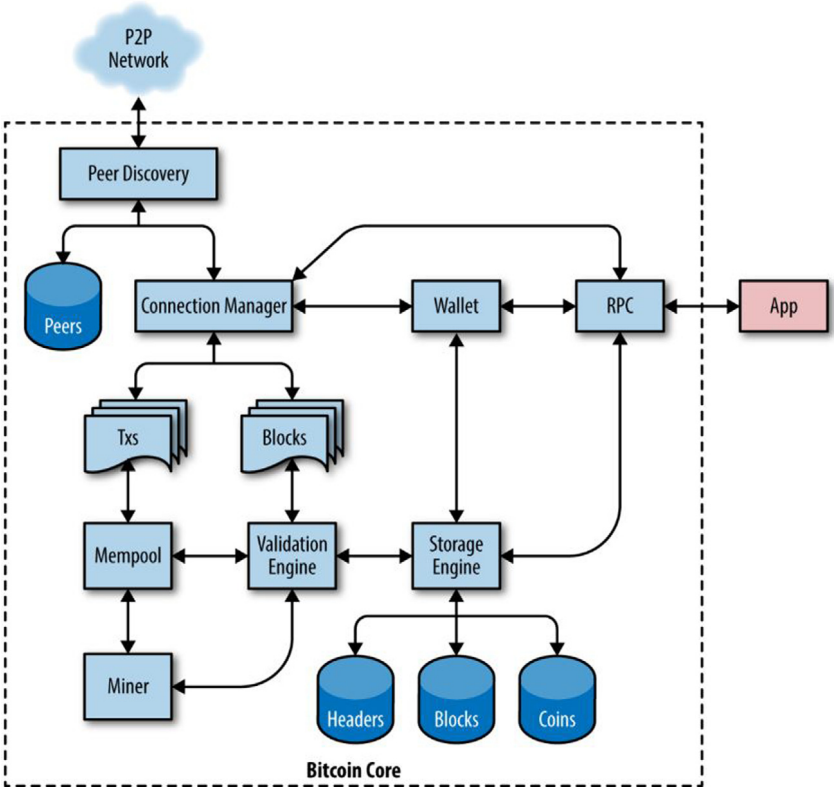

Fig. 2. The reference implementation of the Bitcoin system by E. Lombrozo.

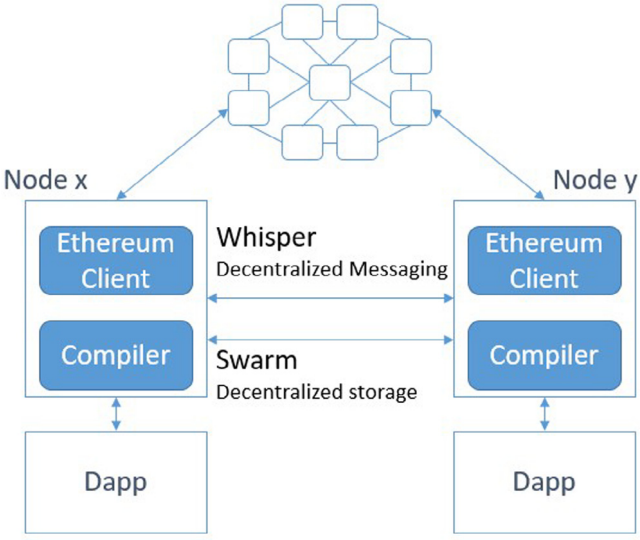

Fig. 3. Ethereum blockchain.

scalability, the cross-shared communication and the computational censorship resistance [28]. In December 2017, the first mobile network infrastructure for Ethereum was introduced [29]. Today's interesting projects prepared by the Ethereum community [30] deal with light clients [31], linking two blockchains channels [32] and allowing off-chain transactions [33].

2.4. Hyperledger

Other communities paved the way for the blockchain technology. In December 2015, the Linux foundation created the Hyperledger project. The objective is the development of the partnership between industries by advancing blockchains [34]. Many companies are working together to support production business networks [35]. Also, Monax, IBM, Soramitsu, R3 and Intel contributed to create Hyperledger Burrow [36], Hyperledger Fabric [37], Hyperledger Iroha [38], Hyperledger Corda [39], Hyperledger Sawtooth [40] platforms respectively. The Hyperledger can be deployed as a private blockchain for business and no consensus is needed. The most notable changes are that peers are now decoupled into two separated run-times with three separated roles: endorser, committer and consenters.

2.5. Algorand

In July 2016, Silvio Micali presented Algorand, a new distributed ledger based on a Byzantine Agreement (BA) [41] that achieves consensus in nine steps without the use of forks. This later pointed to the fact that the consensus can serve others blockchain networks. The main idea in Algorand is that a subgroup of nodes run the consensus on behalf of the entire network while replacing the actors at each step leading to a faster and more secure network [42]. In July 2017, a joint venture between TodaCorp and Algorand is announced in order to scale the blockchain efficiently.

2.6. IOTA

In November 2015, a group of researchers introduced IOTA which relies on an innovative block-less ledger. Essentially, IOTA is characterized by the employment of the tangle technology

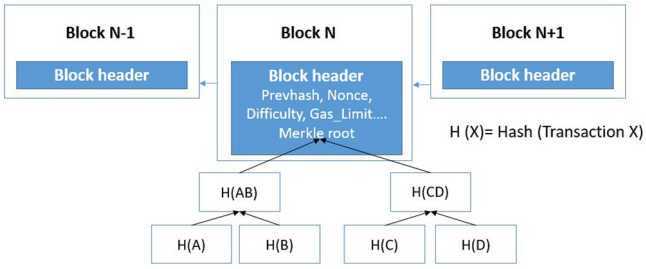

Fig. 4. Blockchain structure.

instead of the blockchain. Besides, no differentiation between the users exists. IOTA is a fee-less currency designed initially for IoT. However, it can be used for micro-payments, machine-to-machine scenarios and scalable applications [43]. In August 2017, IOTA flash network become active enabling instantaneous transactions. Nowadays, many applications involve IOTA [44–47].

3. DLTs concepts

In this section, we overview the key concepts of Bitcoin, Ripple, Ethereum, Hyperledger, Algorand and IOTA architectures comparing them in term of common features, structure and consensus algorithms.

3.1. Common features

Regardless of the technology, DLTs offer numerous security characteristics:

- Immutability: Once added to a block, a transaction become irremovable.
- Auditability: Each block is characterized by cryptographic schemes and secure Timestamping offering the capacity to audit each transaction.
- Integrity: The SIGHASH function validate 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. Also, IOTA relies on Winternitz hash-based cryptography signatures.
- 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 comparison between the several DLTs is summarized in Tables 1 and 2.

3.2. Blockchain structure

The blockchain network is a distributed peer-to-peer environment that works on the concept of trust where no central authority is needed. For that purpose, the different nodes agree on a consensus mechanism to assemble transactions into blocks and append them together to form the structure of the blockchain.

Table 1
DLTs comparison.

Features	Bitcoin	Ripple	Ethereum
Community	Bitcoin developers	Ripple Labs	Ethereum developers
Currency	BTC	XRP	Ether
Consensus	PoW (SHA-256)	RTXP	PoW (Ethash)
Actors	Peers	Peers, Proposals	Peers
Limit	7 tps	1500 tps	20 tps
State	No	No	Data
Block time	10 min	instantly	15 s
Auditing	No	Yes	No
Application	Digital registry, Crypto currency	Crypto currency, Money exchange	Digital registry, Crypto currency, Smart contracts
Smart contract	No	No	Solidity, Serpent, Mutan, LLL
Variants	+ 700 variants	–	Olympic, Frontier, Homestead, Metropolis, Serenity

Table 2
DLTs comparison (Following).

Features	Hyperledger	Algorand	IOTA
Community	Linux foundation	MIT	IOTA Foundation
Currency	None	None	IOTA
Consensus	PBFT	PBFT	PoW (Hashcash)
Actors	Peers, Orderers	Peers, Verifiers	Peers
Limit	No	No	No
State	Key–value	No	No
Block time	Relatif	17 s	–
Auditing	Yes	Yes	No
Application	Digital registry, Smart contracts	Crypto currency	IoT, M2M
Smart contract	Chaincode	No	No
Variants	Burrow, Fabric, Iroha, Corda, Sawtooth	–	–

Cryptographic schemes, specifically Elliptical Curve Digital Signature Algorithm (ECDSA) [48], are used to digitally sign a hash digest of the previous block (N-1) which is used to calculate the hash digest of the current block (N), creating the links between the blocks (see Fig. 4). As an alternative of keeping all transactions information in each block, the merkle tree [18][4] root hash is stored. Each block header contains a nonce, a timestamp, the value of a previous hash, the difficulty[5] and many other parameters and meta-data that differ between the different blockchains (see Fig. 5). For example, the Ethereum block header do not contain only one merkle tree but three trees: a transaction tree to handle transactions, a receipt tree to see the outcome of each transaction and a state tree to handle current balance of each account [30]. Also, Hyperledger and Algorand blocks have a special structure [50] and many fields sorted between block header, block data and block meta-data. Besides, the main difference in a

[4] A hashed based data structure used to verify the data.

[5] Mathematical challenge to prove the correctness of a transaction. In order to meet block goals, the difficulty is adjusted periodically because hashing power and the number of participants in increasing constantly. In a Bitcoin network: New difficulty = Old difficulty ∗ (Actual Time of Last 2016 Blocks/20160 min) [49].

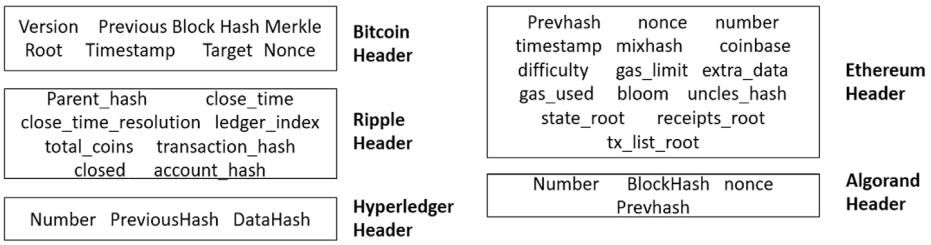

Fig. 5. Blocks headers.

Ripple header consists in the presence of a Boolean flag (closed) which is used to indicate whether the block is settled or not.

3.3. Blockchain networks

The Bitcoin network is an inter-connected network of nodes where each one of them have the same copy of sub-data that we call ledger representing the same blocks that contain the transactions. A transaction is the way you interact with the blockchain. Hence, you submit a transaction into the network via one peer who will forward it to the other peers. In addition to the transactions, each block contains also a hash of the previous block (where the blockchain concept comes from) and as the blockchain will go longer, it will become stronger. Each node will verify if the block is valid and if this is the case, it will be added to the blockchain. The process will restart at each transaction.

In a Ethereum and Hyperledger networks, when a new instance of contract is created, it will be packaged inside a transaction in the payload of the input field. Like any other transactions, it will be hashed and propagates in the network depending on the transaction fees, etc. It will become a part of the block that will circulate. All the receiving nodes will run the transaction which effectively will create a local copy of the same contract in their own internal state.

Besides, the most recent ledger in the Ripple network is referred to as Last Closed Ledger (LCL) and a transaction is any proposed change to the ledger. The goal of the consensus is, for each server, to apply the same set of transactions to the current ledger. Servers continually receive transactions from other servers from the network. These transactions form a candidate set (a pool of transactions waiting to be added to the Ripple Network). At the same time, the server receives proposals from other servers on the network. A proposal is a set of transactions to consider applying to the ledger. The server routes incoming proposals based on a Unique Node List (UNL) which characterizes a list of external servers specific to each node. A proposal from a server not in the UNL is ignored. As a next step, the transactions from the incoming proposal are compared against its candidate set. When a transaction and an incoming proposal matches a candidate set, this latter receives one vote. The server continues to check incoming proposal against its candidate set until a timer expires. At this time, the server takes transaction that have received at least 50% approval rating and packages them into a new proposal which will be broadcasted in the network. This process will be repeated with an approval rate of 60%. Once again, the server receives incoming proposals and looks only at the ones included into its UNL then compares the proposal against its candidate set. Now, transactions with 60% approval rate are packaged inside a new proposal and sent across the network. When the timer expires, the approval rating raises up to 70%. With each generation, the proposals will have a greater similarities to each other and the votes will increasingly agree. The inevitable outcome from this process is that the widely accepted transactions are included. The dissimilarities between proposals are quickly removed. At this point, the network has reached a consensus and 80% of every transaction are either YES or NO [20,51].

In Algorand, a subset of nodes achieves the consensus in nine steps using a modified version of the Byzantine agreement and operate similarly to any PoS algorithm [41]. However, in the Algorand network, the verifiers are self-selected and once the consensus is achieved and the transaction is verified in the 9th step, it is gossiped through the network and added to the blockchain by all peers [52]. The main advantage consists in the capacity to control and corrupt any bad player. Besides, there is no possibility to forge a signature. The idea behind Algorand is that the network suppose that dishonest users are minority. Principally, one random user R is selected and instantly he is identified by the entire network. The user R will assemble a new block of transaction T and broadcasts it on the network. In the other hand, a random committee composed of thousands of users is chosen. This committee runs the block T they have received. If the user R is honest the versions of the block T received would be the same. Therefore, each member of the committee will digitally sign the result (the hash of the block T) and broadcasts the signature in the network. Consequently, if a user A, that is not a member in the committee sees a high percentage of signature for the block T, it assumes that the user is honest and the transaction is permissible, then T will be added to the blockchain. Moreover, in order to prevent any corruption, a different committee is selected in each step [53].

3.4. Permission-less or permissioned network

Bitcoin and Ethereum are permission-less blockchains where anyone can participate in the mining and contribute in blocks verification. To achieve the trust, hashing power is adopted and any miner can create its own block. However, the block will not be added to the canonical chain if it does not encompass a Proof-of-Work[6] (PoW) of a particular difficulty, noting that only the longest chain will be acknowledged. PoW relies on the hardware power to solve the mathematical puzzles leading to the consumption of large amounts of electricity [4]. Moreover, miners with extra dominant hardware have more successful rate to solve the challenge and get the rewards. However, owning 51% of network power will lead to control the network. And that is the reason behind finding an alternative to the PoW algorithm in the new release of Ethereum. The Ethereum Serenity release is considering switching to the Casper [4] protocol which is a Proof-of-Stake (PoS) algorithm where the validators (i.e. the peers who validate the blocks) hold a "stake" of a minimum of 1000 Ethers as a security-deposit [4]. In the PoS algorithm, the validators are rewarded if they use their stake correctly otherwise they are penalized. Another advantage of this approach is that is faster to a validator to bet on a block compared to a miner to solve the

[6] An algorithm to prove the achievement of a certain amount of work.

cryptographic puzzle in order to append the block. Therefore, the whole blockchain will improve and become faster.

As for the Hyperledger, it is considered a permissioned network. The participants necessitate to be recognized although not inevitably fully trusting each other. All the parties have their own copy of the distributed ledger and only see specific transactions related to their businesses. A known community build a system in common on top of a shared ledger and avoid running a PoW consensus mechanism. Only specific actors are able to perform the validations and nodes are divided into peers (execution nodes) and orderers (nodes that agree on blocks order). After the client submits a transaction to the endorsing peer, the latter harvests an endorsing signature for the client to submit it through the ordering service which sends the identification sequence number to the peers. Hence, all peers commit and apply the same succession of transactions and update their sequence of transactions and will have the same state [54].

Ripple is also a permissioned network giving the existence of a default list (UNL) that needs to be considered in the validation process. Besides, the majority of the validating servers are operated by Ripple Labs.

As for Algorand, it can be positioned as permissioned or permissionless network. The users and the members of the committee are identified immediately when running the consensus.

In IOTA, a new distributed ledger concept is employed. While all the previous DLTs are based on the blockchain technology, IOTA relies on Tangle utilizing directed acyclic graph (DAG) technology (see Fig. 6).

3.5. The tangle technology

In order to operate at a higher rate, remove mining charges and eliminate transaction issuers and approvals, the tangle technology was introduced. The main concept is when a transaction is created, two others transactions are automatically validated. The tangle technology is used to designate IOTA's DAG and is a simplification of the blockchain technology which define a special case of DAG [55]. A DAG involves no directed cycle. The vertex never follows the same paths and each one is only comprised once which means that there is no return to the same vertex. There is no mining in IOTA and peers can function without being connected to the main tangle network. Moreover, unlike others DLTs, IOTA uses quantum resistant algorithms [55]. While the transactions are stored in blocks in blockchains networks, they are bundled in edges in DAG. Each account grips only its balance history. Furthermore, the ordering of transactions is performed asynchronously. Three steps are needed for each transaction (See Fig. 8):

1. The node creates a transaction and signs it with the private key.
2. The transaction goes in tip selection mode. Using Random Walk Monte Carlo (RWMC) [56], the node picks two previously unconfirmed transactions (tips) (see Fig. 6) to confirm them against the tangle history by solving a PoW cryptographic challenge.
3. Once confirmed, the transaction becomes part of the tangle and part of the branch.

As a way of minimizing the tangle size, a snapshot can be created by removing the records with zero balance.

3.6. 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 [17] because the peers are divided into endorsers, committers and consenters. Also, the parallel transactions processing in the Hyperledger blockchain leads to a higher throughput. 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 [57], bidirectional channels were presented. As for the Ripple network which runs a Byzantine agreement, the validators nodes only have a copy of the LCL and do not include the history of all transactions. In addition, the XRP ledger is settled every 4 s. Consequently, the Ripple network does not experience the aforementioned drawbacks and can scale up to 1500 tx/s. Besides, many functions in an Ethereum network need to be replicated between users causing scalability concerns [17]. Thus, the introduction of state channels and plasma channels in Ethereum networks to perform some transactions offline and conduct off-chain activities leading also to enforce anonymity. Furthermore, in [58], the sharding techniques which consist in splitting accounts' states into separated chunks, were introduced. In the other hand, IOTA's network is more effective when the number of transactions become higher. Transactions confirmations result in 112 tx/s for 250 nodes [59]. As for Algorand's scalability, it is on demand and managed by a new Byzantine Agreement. Also, using the TODA framework [60], this network can scale greater than 3M tx/s and 4B nodes. Fig. 7 represents DLTs network scalability performance. Assuredly, permissioned networks operate more efficiently than the permissionless environments. Besides, PoS algorithms scale better.

Table 3
DLTs known attacks.

DLT	Attacks
Bitcoin	Majority Attack DDOS - Coins Thefts - Wallets attacks - Sybil attack - Traffic sniffing - Energy Consumption - Clock attack - Forks - Double-Spending attack - Spam attack - Keys Theft - Fee snipping - SPV attacks
Ripple	Wallets attacks - Gateways attack - Thefts - Incentive Misalignment
Ethereum	Majority Attack - Call to the unknown - Exception disorder - Gasless send - Types costs - Reentrancy - Secret attack - Immutable bugs - Unpredictable state - Generating randomness - Time constraints - DDOS - Coins Thefts - Double-Spending attack - Spam attack - Forks - Snapshots attachments - Keys Theft - Eclipse attack
Hyperledger	Spam attack - Consensus attack - Chaincode errors
Algorand	Algorand is not widely adopted (still in the testing mode)
IOTA	Majority Attack - Transactions spamming - Coins Thefts - Double-Spending attack - Snapshots attachments - Seeds generation - Wallets attacks - Keys Theft

4. DLTs security challenges

The main concept in blockchains is decentralization and it has imperatives allegations on the security (Table 3). 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.

4.1. Anonymity

Metadata exposure disturbs the participant's anonymity and reduce confidentiality. Besides, being aware of a user's address

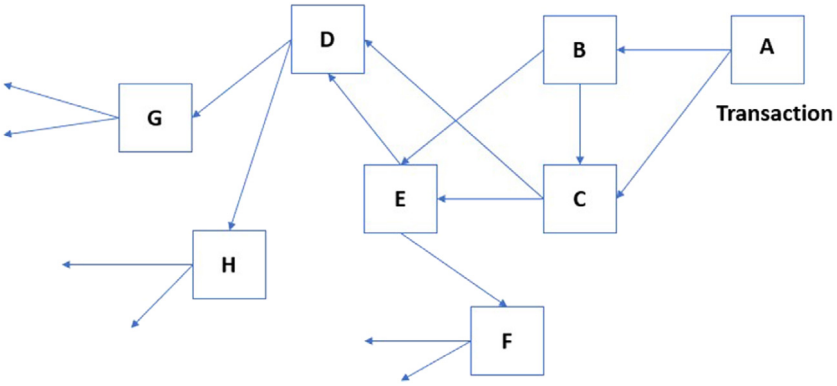

Fig. 6. IOTA network.

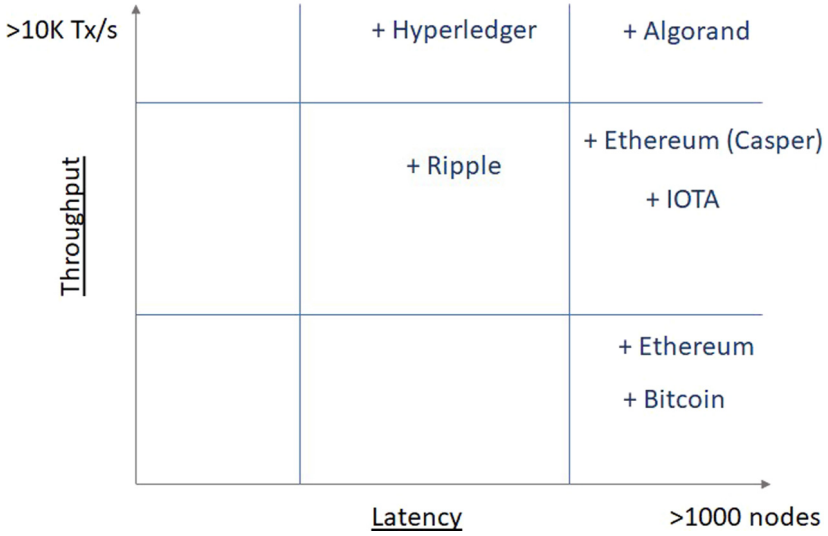

Fig. 7. Network scalability.

will lead to the history and exchanges tracking for financial and non-financial applications. Thus, the implementation of 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 dis-adequately 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 [61]. Whereas, for a few applications, it takes 1 h to forge a proof. Besides, pseudonyms (Wallets-IDs) which are required for authentication, present another measure for anonymity as long as they are not linked to an entity. However, if a pseudonym becomes known, the entire history can be revealed.

4.2. Consensus

Several consensus algorithms [17] 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 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 [62] considering the evolution of computing power related to parallel architectures (GPGPU, MPPA, multicore ...) as well. Besides, if the network dramatically loses mining computation power, block creation decelerates widely, especially against a targeted DDoS attack against the actors with the most PoW. 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 [63] aims, from a particular reward value, to increase stackers affluence over time even if the bet has not changed, allowing them to gain money 33% faster. Regarding the Ripple consensus, the validating servers are mainly owned and controlled by Ripple Labs. Therefore, if these terminals are compromised, the network is at risk. Furthermore, Ripple Labs holds a significant portion of XRPs which effectively regulate the economy. 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 [64]. Therefore, it depends on

stateless application only. Furthermore, applications relying on a strong cryptography are not reinforced allowing opponent access to some replicas [64]. In the other hand, the introduction of a new byzantine agreement that achieves the consensus in nine steps in the Algorand network, allows by some means a secure consensus specially with the randomization of committees and the replaceability of members.

4.3. 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 [65], and in some scenario not irreversible. The introduction of multi-signature addresses and cloud storage systems may enable damage resistance unless collusion.

4.4. Smart contracts vulnerabilities

Numerous causes make smart contracts principally susceptible to errors [17]. The first reason is related to the vulnerabilities in the high-level programming language supported, specifically the solidity language in Ethereum.

Moreover, the fact that a transaction is irremovable or unchangeable, it will be challenging to remove the code security flaws once added to the network. The solutions will consist in auditing smart contract code security primarily and adding expiration date to the Dapps secondly. Furthermore, publishing newly discovered system weaknesses will let the hacker take advantages of these vulnerabilities and conduct unpredictable fraudulent activities. Besides, some DLT features are not permanently useful expressly when the blockchain network is under attack. As an example, the case of an attacker writing a smart contract with an infinite loop: the transaction is valid, however, if the attacker is able to pay the gas fees to maintain the exchange, the miner will be the gainer. Also, Ether may be lost in the transfer if the address of the recipient is not valid. Many resolutions for these problems have been applied by the Hyperledger technology. For instance, the validation and auditing of smart contracts in order to remove any potential system exploitation.

4.5. Transactions security

Blockchain security depends on cryptographic schemes and is based on network's management over keys. 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 [66]. Moreover, a more recurrent example of exploitation consists in controlling the target user inbound and outbound connections by implementing an eclipse attack [67]. In [68], a double-spending attack was performed during block forks where typically in similar scenario the longest chain is adopted.

4.6. 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 [69]. The tightening of time range and the usage of peer's system timing or Blockchain median time, might be possible solutions.

4.7. 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 peer's 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 takes 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.

4.8. Cryptography contravention

ECDSA [48] and secp256k1 [70] that are applied in Bitcoin, Ethereum, Ripple, Hyperledger and Algorand, 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. Besides, in [71], lattice attacks against weak ECDSA signatures in cryptocurrencies have been presented.

For IOTA's DLT, the Winternitz one time signature scheme (W-OTS) [72] is applied. W-OTS uses shorter signatures and is relatively considered quantum robust. IOTA seeds should be created with cautions using trusted generators.

4.9. Snapshot attachment

A snapshot consists in compiling unspent transactions in order to bootstrap a new blockchain. Attaching a snapshot to the network may have huge security implications, specifically when reusing the same address in outgoing transactions. Moreover, after a snapshot, the wallet will use the same address to recreate the history. The use of a new wallet with a new address can be adopted as a countermeasure.

4.10. Targeted DDoS attacks

These attacks target the most critical actors and consist in flooding the network by lots of information in a way it becomes irresponsive to transactions [73]. 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 clients' protections related to the amount of check-signature, block size limitation, orphan transactions banning and ignorance of not standard transactions.

5. Attacks

This section exposes the results of three DLTs attacks that we have simulated: IOTA spamming to improve and accelerate the number of approved transactions, a majority attack targeting Bitcoin and a reentrancy attack leveraging a malicious smart contract in Ethereum.

```
Worker 2: Preparing transactions
Worker 2: Requesting transactions to create confirmations for.
Worker 2: Waiting for job vacancy in worker pool.
Worker 1: Completed PoW (Proof of Work), broadcasting confirmations.
Worker 0: Performing PoW (Proof of Work)
Worker 1: Broadcast completed.
New transaction created: thetangle.org iotasear.ch iota.tips
Worker 1: Preparing transactions
Worker 1: Requesting transactions to create confirmations for.
Worker 1: Waiting for job vacancy in worker pool.
Worker 0: Completed PoW (Proof of Work), broadcasting confirmations.
Worker 2: Performing PoW (Proof of Work)
Worker 2: Completed PoW (Proof of Work), broadcasting confirmations.
Worker 1: Performing PoW (Proof of Work)
Worker 0: Broadcast completed.
New transaction created: thetangle.org iotasear.ch iota.tips
Worker 0: Preparing transactions
Worker 0: Requesting transactions to create confirmations for.
Worker 2: Broadcast completed.
```

Fig. 8. IOTA transactions confirmation - spammer.

```
2018-07-29T08:42:45.952Z: Randomly changing IOTA nodes
2018-07-29T08:42:45.952Z: Changing nodes to balance the load
2018-07-29T08:42:45.952Z: New transaction created with hash: BGVFUKUPEZGQHFYUB9GDCPIPXZYPJFPUMFAWNQSNAWTTT9TJFQLGXAEVCQMLEIM9JHEYQQVFCBMVZ9999
2018-07-29T08:42:45.951Z: View transaction details at: open-iota.prizziota.com - iotasear.ch - thetangle.org
2018-07-29T08:42:45.951Z: Completed PoW (Proof of Work) on 2 transactions
2018-07-29T08:42:27.665Z: Performing PoW (Proof of Work) on 2 transactions
2018-07-29T08:42:27.665Z: Node is synced
```

Fig. 9. IOTA spammer: Nodes load balancing.

Table 4
IOTA spammer simulation using load balancing.

TXs added to the tangle	Approved TXs/min	TXs added/min
1	7.4737	3.737
2	10.600	5.3
3	17.699	8.85
4	17.825	8.912

Table 5
IOTA Spammer simulation without load balancing.

TXs added to the tangle	Approved TXs/min	TXs added/min
1	1.514	0.757
2	1.355	0.677

5.1. 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 losing gas and computation power. This results in decreasing of the number of reachable peers and entire network outage [74]. This holds true for all DLTs except IOTA.

In IOTA, the network becomes faster when the number of transactions increases. In order to speed up transactions confirmations, we can spam the network with transactions [75]. For that purpose, we used the spammer developed by Peter Ryszkiewicz available on Github [75] with the WebGL framework in order to utilize the GPU from a web application [76]. The results of our simulation are summarized in Table 4 . Definitely the network quality has an impact on confirmations. The first set of spamming tests were performed using load balancing: the spammer will change node after each transaction (Fig. 9). We have conducted also the same simulations without nodes load balancing. The number will decrease to 1.514 approved transactions per minute for the first transaction added to the tangle and to 1.355 approved transactions per minute when 2 transactions are added to the tangle (Table 5).

The results show that, in order to have a successful IOTA spamming attack that increases the number of approved TXs added to the tangle per minute, nodes' load balancing is needed.

```
mapping (address => uint) private userBalances;

function withdrawBalance() public {
    uint amountToWithdraw = userBalances[msg.sender];
    require(msg.sender.call.value(amountToWithdraw)());
    userBalances[msg.sender] = 0;
}
```

Fig. 10. Insecure code.

5.2. Malicious contracts

A smart contract does not deal with exceptions, like for example when the transactions are recorded but unpredictably restructured or postponed [77]. For instance, we simulated a Reentrancy attack [78] using Ganache [79] that provides a web interface for the truffle framework, installs on our machine a private blockchain network, and interacts with our Ethereum wallet where our malicious smart contract is created and signed. The idea is to let a contract call back into a function before conditions are updated. As shown in Fig. 10, we can call into the function and succeed because the balance is set at the end of the function. The balance is being sent back without deduction. Therefore, recalling the withdraw will deduct the balance from the total until extracting the whole amount. Using the call function will invoke

ADDRESS
0×f17f52151EbEF6C7334FAD080c5704D77216b732

BALANCE
99.99 ETH

Fig. 11. Reentrancy transaction.

SENDER ADDRESS
0×f17f52151ebef6c7334fad080c5704d77216b732

CREATED CONTRACT ADDRESS
0×b9a219631aed55ebc3d998f17c3840b7ec39c0cc

CONTRACT CREATION

VALUE
0.00 ETH

GAS USED
139192

GAS PRICE
40000000000

GAS LIMIT
139192

MINED IN BLOCK
1

TX DATA

0×6060604052341561000f57600080fd5b6101468061001e6000396000f300606060405260043610610041576000357c0100000000000000000000000000000000000000000000000000000000
000000900463ffffffff1680633ccfd60b14610046575b600080fd5b341561005157600080fd5b61005961005b565b005b3373ffffffffffffffffffffffffffffffffffffffff166108
fc60008033731ffffffffffffffffffffffffffffffffffffffff1673ffffffffffffffffffffffffffffffffffffffff16815260200190815260200160006040518115029060405160006
0405180830381858888f193505050501561011857600080600033731ffffffffffffffffffffffffffffffffffffffff1673ffffffffffffffffffffffffffffffffffffffff16815260
200190815260200160006020819055505b5600a165627a7a72305820fb509ed6c35bcb5e8f6c0002525866a7ba654b579cd6362e4f03bd06fd84e4b50029

Fig. 12. Reentrancy problem.

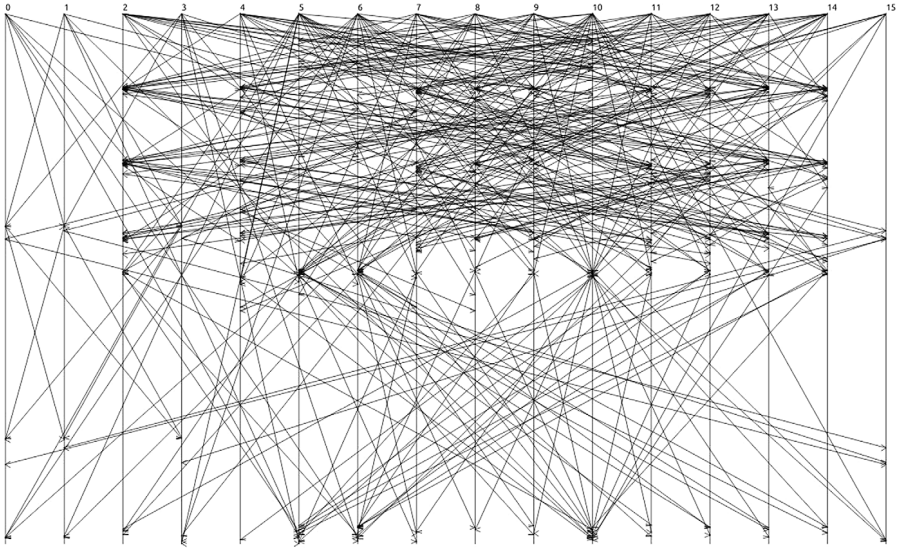

Fig. 13. Netanim simulation.

the interaction with the main contract several times before its execution is completed, causing bugs [80]. As shown in Figs. 11 and 12, 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 loses all his funds or until the maximum call stack depth is reached.

5.3. 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 adjusted the Bitcoin simulator [81], presented in [82], in order to simulate a majority attack targeting the Bitcoin network. This attack permits to take control over the mining hashing power and thus conducting other malevolent behaviors.

The simulation involved 16 nodes (see Fig. 13). For the several tests conducted, we adjusted the hashing rate for each node in the *minerHash* table of the simulator. The attacker's mining percentage is the last field of *minerHash*. The remainder of the input parameters are configured using their default values (*blockSize* = −1, minConnections = −1, maxConnections = −1, invTimeoutMins = −1).

Table 6
Majority attack simulation.

Hashing power rate	Nb of confirmations	Probability of attack
0.1	5	0.00091368
0.1	6	0.0002428
0.2	5	0.0274155
0.2	6	0.0142506
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

The results of our simulation are summarized in Table 6. The Probability of controlling the mining consortium is dependent of the hashing power rate and the number of confirmations.

Probability of Control (*Hashing power rate*, *Nb of confirmations*) = P.

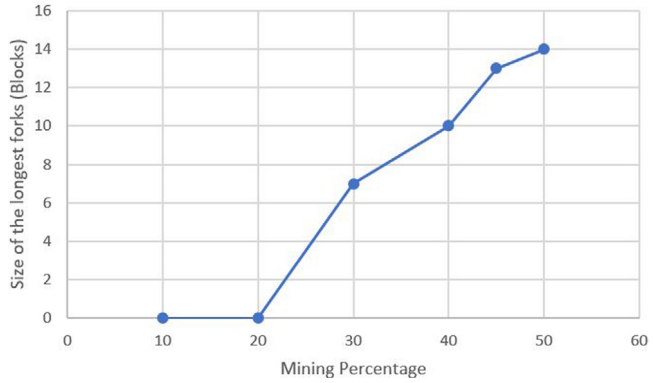

Fig. 14. Longest fork size.

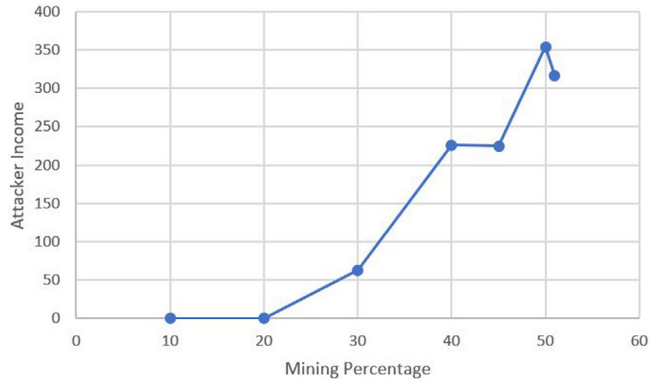

Fig. 15. Attacker income.

Having the same hashing power rate, we demonstrated that a lower amount of confirmations will lead to a higher probability of control and risk level (see (1) and (2)).

$$\textit{Probability of Control}\,(0.4, 3) = 0.664168. \tag{1}$$

$$\textit{Probability of Control}\,(0.4, 4) = 0.603401. \tag{2}$$

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.

$$\textit{Probability of Control}\,(0.5, \textit{Nb of confirmations}) = 1.$$

This type of attack is a consequence of other malevolent scenarios such as stealing user machine power through trojan system exploitation [83], use timestamps dependency attack to manipulate the mining or add Call stack depth limit function and loops that can trigger gas limit.

Fig. 15 shows that the attacker income increases with the growth of the percentage of the selfish mining. Also, as shown in Fig. 14, the number of blocks included in a fork is larger when the miner hashing power is higher.

Selfish mining is also recurrent in IOTA as well. In [84], the authors proved that tips selections and transactions attachments can be optimized.

6. Conclusion

In this paper, we compared five DLTs networks and introduced the related consensus algorithms used to achieve agreement. Our study included Bitcoin, Ripple, Ethereum, Hyperledger and Algorand which rely on the Blockchain technology. On the other hand, we brought IOTA DLT that is based on Tangle onto the analysis. This technology is designed specifically for IoT and utilizes a new bloc-less architecture for transactions' confirmation. However, the basis of both frameworks remains the same. Both technologies require N verifications based on the Monte Carlo simulation in IOTA and on PoW and BA in the others.

We discussed also their security challenges and proposed some countermeasures to dissuade flaws in these networks. Moreover, we simulated a majority attack in the Bitcoin network, a reentrancy attack using a malicious smart contract in Ethereum, and we spammed IOTA network with transactions in order to speed up the number of confirmations in the network.

Our paper provides a comprehensive comparison between DLTs focusing on their security challenges. Furthermore, we overview their scalability, consensus issues and peers' fairness.

Lots of industries are stirring to create services into the blockchain but on the other hand, the data pollution or misuse of the data is an immense problematic [85]. Blockchain can also be utilized by malicious entities and for illegal use cases [11]. Recently, attackers misused the Tor network and leveraged the blockchain to conduct several malicious behaviors [86]. Besides, the blockchain network can be used for malware distribution by submitting malicious content to be encrypted using public key cryptography and selling the private key for decrypting the payload and abuse it other than crypto-jacking [87] and money laundering attacks [88]. Moreover, the bloating problem of the blockchain networks will most certainly lead to several forms of badly behaved mechanisms because the expansion of the network will leave it, by some means, uncontrolled. Another area of misuse, consists in letting the blockchain network act as Command and Control Servers (CnC) to the infected bots to receive additional instructions or malicious code. Recently, a new form of hacking consists in using user's environments to mine cryptocurrencies and increase their own rewards without the victim's knowledge [89].

Our future work will include broadening our scope to study other new generation DLTs performance. In addition, we will tackle their malicious applications.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

[1] R. Wattenhofer, Distributed Ledger Technology - The Science of Blockchain, Forest Publishing, 2017.
[2] R. blog, Top 100 blockchain organisations: From coindesk to bitpay, these are the most influential organisations in the distributed ledger space, 2018, https://richtopia.com/top-lists/top-100-blockchain.
[3] A. Rosic, 5 blockchain applications that are shaping your future, 2017, https://www.huffingtonpost.com/ameer-rosic-/5-blockchain-applications_b_13279010.html.
[4] A. Bahga, V. Madisetti, Blockchain Applications: A Hands-on Approach, VPT, 2017.
[5] R. AlTawy, M. ElSheikh, A.M. Youssef, G. Gong, Lelantos: A Blockchain-Based Anonymous Physical Delivery System, Tech. rep., Cryptology ePrint Archive, Report 2017/465, 2017, http://eprint.iacr.org/2017/465.
[6] A. Rosic, 6 blockchain applications that go beyond bitcoin, 2016, https://due.com/blog/6-blockchain-applications-go-beyond-bitcoin/.
[7] J. Sagar, How will blockchain change the way we trade online? 2017, https://www.newsbtc.com/2017/12/14/blockchain-change-online-trade/.
[8] S. Higgins, Bitcoin exchange youbit to declare bankruptcy after hack, 2017, https://www.coindesk.com/south-korean-bitcoin-exchange-declare-bankruptcy-hack/.

[9] J. Moubarak, M. Chamoun, E. Filiol, Developing a k-ary malware using blockchain, in: NOMS 2018-2018 IEEE/IFIP Network Operations and Management Symposium, IEEE, 2018, pp. 1–4.
[10] J. Moubarak, E. Filiol, M. Chamoun, On blockchain security and relevant attacks, in: Communications Conference (MENACOMM), IEEE Middle East and North Africa, IEEE, 2018, pp. 1–6.
[11] T. Fox-Brewster, Bitcoin's blockchain offers safe haven for malware and child abuse, warns interpol. forbes, 2015.
[12] L. Lamport, The part-time parliament, ACM Trans. Comput. Syst. (TOCS) 16 (2) (1998) 133–169.
[13] L. Lamport, et al., Paxos made simple, ACM Sigact News 32 (4) (2001) 18–25.
[14] E.A. Brewer, Towards robust distributed systems, in: PODC, Vol. 7, 2000.
[15] A. Fox, 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.
[16] S. Gilbert, N. Lynch, Brewer's conjecture and the feasibility of consistent, available, partition-tolerant web services, Acm Sigact News 33 (2) (2002) 51–59.
[17] J. Moubarak, E. Filiol, M. Chamoun, Comparative analysis of blockchain technologies and TOR network: Two faces of the same reality? in: CSnet , 1st Cyber Security in Networking Conference, IEEE, 2017.
[18] S. Nakamoto, Bitcoin: A peer-to-peer electronic cash system, 2008.
[19] C. Decker, R. Wattenhofer, Information propagation in the bitcoin network, in: Peer-to-Peer Computing (P2P), 2013 IEEE Thirteenth International Conference on, IEEE, 2013, pp. 1–10.
[20] D. Schwartz, N. Youngs, A. Britto, et al., The ripple protocol consensus algorithm, in: Ripple Labs Inc White Paper, Vol. 5, 2014.
[21] A. Ghosh, M. Mahdian, D.M. Reeves, D.M. Pennock, R. Fugger, Mechanism design on trust networks, in: International Workshop on Web and Internet Economics, Springer, 2007, pp. 257–268.
[22] P. Moreno-Sanchez, A. Kate, M. Maffei, K. Pecina, Privacy preserving payments in credit networks, in: Network and Distributed Security Symposium, 2015.
[23] V. Buterin, et al., A next-generation smart contract and decentralized application platform, 2014, white paper.
[24] H. Diedrich, Ethereum: Blockchains, Digital Assets, Smart Contracts, Decentralized Autonomous Organizations, Wildfire Publishing, 2016.
[25] G. Wood, Ethereum: A secure decentralised generalised transaction ledger, in: Ethereum Project Yellow Paper, Vol. 151, 2014.
[26] G. Wood, DEVCON1: Shh! Whisper - Gavin Wood.
[27] V. Tron, A. Fischer, D. Nagy, Z. Felfld, N. Johnson, Swarm, etherspher, 2016.
[28] V. Buterin, Ethereum 2.0 mauve paper, 2016.
[29] G. Network, From the winning team at ethwaterloo world's largest ethereum hackathon, 2017.
[30] C. Dannen, Introducing ethereum and solidity.
[31] Github, https://github.com/paritytech/parity/wiki/Light-Client.
[32] J. Coleman, State channels.
[33] Github, https://github.com/pipermerriam/ethereum-computationmarket.
[34] Hyperledger foundation, https://www.hyperledger.org/about/join.
[35] IBM, Building a blockchain for business with the hyperledger project.
[36] https://github.com/hyperledger/burrow.
[37] https://github.com/hyperledger/fabric.
[38] https://github.com/hyperledger/iroha.
[39] https://github.com/corda/corda.
[40] https://github.com/hyperledger/sawtooth-core.
[41] Y. Gilad, R. Hemo, S. Micali, G. Vlachos, N. Zeldovich, Algorand: Scaling byzantine agreements for cryptocurrencies, in: Proceedings of the 26th Symposium on Operating Systems Principles, ACM, 2017, pp. 51–68.
[42] Coindesk, Scaling consensus? this turing winner thinks he's found a way, 2017.
[43] reddit, Iota, 2015.
[44] D. Pollock, Micropayment company ditches "outdated bitcoin" for iot technology, 2017.
[45] F. d'Anconia, Iota blockchain to help trace families of refugees during and after conflicts, 2017.
[46] A. Mandelli, Iota plans for the future, announces major partnerships, 2018.
[47] J. Keane, Trust your odometer? blockchain test aims to turn tide on car tampering, 2017.
[48] E. Rykwalder, The math behind bitcoin, 2014, https://www.coindesk.com/math-behind-bitcoin/.
[49] A.M. Antonopoulos, Mastering Bitcoins, O'Reilly Media, Sebastopol, 2014.
[50] Hyperledger fabric blog, http://blockchain-fabric.blogspot.com/2017/04/hyperledgerfabric-v10-block-structure.html.
[51] Steemit, Understanding xrp ledger with pseudocode, 2017, https://steemit.com/ripple/@tabibito567/understanding-xrp-ledger-by-pseudocode.
[52] J. Chen, S. Micali, Algorand: the efficient and democratic ledger, 2016, CoRR abs/1607.01341.
[53] "Algorand: A better distributed ledger", with Silvio Micali, https://www.youtube.com/watch?v=_nQE_HAGlmM.
[54] Hyperledger blog, http://hyperledger-fabric.readthedocs.io/en/latest/arch-deepdive.html.
[55] S. Popov, The tangle, IOTA (2016).
[56] W.-J. Cheng, J. Cox, P. Whitlock, Random walks on graphs and Monte Carlo methods, Math. Comput. Simulation (2015).
[57] C. Decker, R. Wattenhofer, A fast and scalable payment network with bitcoin duplex micropayment channels, in: Symposium on Self-Stabilizing Systems, Springer, 2015, pp. 3–18.
[58] M. Scherer, Performance and scalability of blockchain networks and smart contracts, 2017.
[59] Mobilefish.com, 2017.
[60] TodaCorp, Toda is a layer-zero blockchain protocol, 2017.
[61] V. Butterin, Zk-snarks: Under the hood, 2017.
[62] S. Barber, X. Boyen, E. Shi, 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.
[63] M. Uddin, Attacking casper at ethereal hackathon, 2017.
[64] N. Chondros, K. Kokordelis, 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.
[65] Is the official light wallet hacked?, https://github.com/iotaledger/wallet/issues/430th.
[66] A. Gervais, H. Ritzdorf, G.O. Karame, 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.
[67] E. Heilman, A. Kendler, A. Zohar, S. Goldberg, Eclipse attacks on bitcoin's peer-to-peer network, in: USENIX Security Symposium, 2015, pp. 129–144.
[68] A. Gervais, H. Ritzdorf, G.O. Karame, Double-spending fast payments in bitcoin due to client versions 0.8. 1, 2013.
[69] C. S. Community, Bitcoin under attack!!, 2017, https://securitycommunity.tcs.com/infosecsoapbox/articles/2017/11/04/bitcoin-under-attack.
[70] H. Mayer, Ecdsa security in bitcoin and ethereum: a research survey, 2016.
[71] J. Breitner, N. Heninger, Biased nonce sense: Lattice attacks against weak ECDSA signatures in cryptocurrencies, IACR Cryptol. ePrint Arch. 2019 (2019) 23.
[72] A. Hülsing, W-ots+–shorter signatures for hash-based signature schemes, in: International Conference on Cryptology in Africa, Springer, 2013, pp. 173–188.
[73] R.R. O'Leary, Bitcoin gold website down following ddos attack, 2017, https://www.coindesk.com/bitcoin-gold-website-following-massive-ddos-attack/.
[74] L. Parker, Bitcoin 'spam attack' stressed network for at least 18 months, claims software developer, 2017, https://securitycommunity.tcs.com/infosecsoapbox/articles/2017/11/04/bitcoin-under-attack.
[75] pRizz, Iota spammer webapp, 2017, https://securitycommunity.tcs.com/infosecsoapbox/articles/2017/11/04/bitcoin-under-attack.
[76] GpanosXP, Iota spammerxp, 2017.
[77] N. Atzei, M. Bartoletti, 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.
[78] Github, 2016. https://github.com/ethereum/wiki/wiki/Safety.
[79] Working with ganache, http://truffleframework.com/docs/ganache/using.
[80] D. Wong, Attacks on ethereum smart contracts, 2017.
[81] A. Gervais, Bitcoin simulator, 2017, http://arthurgervais.github.io/Bitcoin-Simulator/index.html.
[82] A. Gervais, G. Karame, K. Wüst, V. Glykantzis, H. Ritzdorf, S. Capkun, On the security and performance of proof of work blockchains, in: Proceedings of the 23nd ACM SIGSAC Conference on Computer and Communication Security (CCS), ACM, 2016.
[83] K. Huang, Security 101: The impact of cryptocurrency-mining malware, 2017.
[84] S. Popov, O. Saa, P. Finardi, Equilibria in the tangle, 2017, arXiv preprint arXiv:1712.05385.
[85] S.T. Ali, P. McCorry, P.H.-J. Lee, F. Hao, Zombiecoin: powering next-generation botnets with bitcoin, in: International Conference on Financial Cryptography and Data Security, Springer, 2015, pp. 34–48.
[86] T. Fox-Brewster, How Hackers Abused Tor to Rob Blockchain, Steal Bitcoin, Target Private Email and Get Away With It, Forbes, 2015.
[87] N. Communications, Cryptojacking: The Dark Side Of Blockchain, NEMA, 2018.
[88] T.K. Sharma, How does bitcoin money laundering work?, 2018, https://www.blockchain-council.org/blockchain/how-bitcoin-money-laundering-works/.
[89] J. Pearson, At Least 1.65 Million Computers Are Mining Cryptocurrency for Hackers So Far This Year, Motherboard, 2017.

Joanna Moubarak is a cybersecurity researcher, a peer reviewer and a lecturer. Currently, she is heading the Potech Labs. She holds a Master degree in Network and Systems Security and a PhD in "Computer Science and Networks" from Saint Joseph University of Beirut (USJ). Mrs. Moubarak has worked in multiple cybersecurity domains as an IT Security Consultant, overseeing different security solutions and controls as well as maintaining business continuity. Her research studies deal with networks security, malware analysis and Blockchain. Joanna has multiple publications in refereed journals and conferences on malware and distributed ledgers technologies.

She has more than 7 years' experience in IT security consulting helping enterprises building governance, risk and compliance, automating business processes and managing corporate risk. She was involved in several integration projects and has worked on multiple security solutions including AirTight, Airwatch, FireEye, ForeScout, Intel Security, F5, Kaspersky and Palo Alto. She is specialized in Mobile Device Management, Next-Generation Firewalls, Advanced Threat detection, Data Loss Prevention as well as SIEM solutions. Furthermore, she teaches graduate students courses related to Network management and Security, Cybersecurity Ethics and Laws, Blockchain and Computer Virology.

Maroun Chamoun is a professor at the Faculty of Engineering of Saint Joseph University (USJ) in Beirut. He holds a degree in Computer Engineering from Saint Joseph University and a Master in Intensive Calculus from the joint program between the Lebanese University (UL) and Saint- Joseph University of Beirut. He holds a PhD in "Computer Science and Networks" from Telecom ParisTech in Paris. He teaches "ethical hacking", "Malware Analysis", "operating systems", and "language theory and compilers". He is a member of the research center CIMTI (Center for Computer Science, Modeling and Information Technologies) where he was the director between 1998 and 2015. His research interests include Cybersecurity mainly in Virology and Threat Detection, Cryptography mainly Homomorphic Schemes, Operating Systems mainly in scheduling and system protection, Compilers and Computer Languages mainly in correctness proof of compilers. He has more than 20 journal and conference publications in the domain of Cybersecurity.

Eric Filiol is professor at ENSIBS, Vannes, France and at National Research University Higher School of Economics, Moscow, Russia in the field of information and systems security. He is also a senior consultant in cyber security and intelligence. He directed the research and the cyber security laboratory of a French engineer school for 12 years. He spent 22 years in the French Army (Infantry/French Marine Corps). He holds an engineering degree in Cryptology, a PhD in Applied Mathematics and Computer Science from Ecole Polytechnique and a Habilitation to Conduct Research (HDR) in Information from the University of Rennes. He holds several NATO certifications in the field of intelligence. He is editor-in-chief of the research journal in Computer Virology and Hacking Techniques published by Springer. He regularly gives international conferences in the field of security (Black Hat, CCC, CanSecWest, PacSec, Hack.lu, Brucon, H2HC...).